
Generación de narrativas en museos virtuales



Trabajo Fin de Grado en Ingeniería del Software

Juan José Arconada Alonso
Roberto Moreno Rojas
Víctor Díaz Yagüe

Dirigido por

D. Gonzalo Rubén Méndez Pozo
D^a. Raquel Hervás Ballesteros

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid

Madrid, 1 de Septiembre de 2017

“La realidad le deja mucho a la imaginación”

- JOHN LENNON

“Vivimos en una sociedad exquisitamente dependiente de las ciencias y la tecnología, en la cual prácticamente nadie sabe nada acerca de la ciencia o la tecnología”

- CARL SAGAN

“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber”

- ALBERT EINSTEIN

Agradecimientos

Juan José Arconada Alonso

"Cuando la gratitud es tan absoluta las palabras sobran"

- ÁLVARO MUTIS

A mis profesores. A mis compañeros. A los que estuvieron, están y estarán. A los que estuvieron y ya no están. Todos sabéis quiénes sois. Gracias.

Roberto Moreno Rojas

En primer lugar, mis agradecimientos van dedicados a nuestros tutores de este trabajo, D. Gonzalo y D^a. Raquel, por guiarnos durante todo este año, ayudarnos a alcanzar nuestros objetivos y adaptarse constantemente a nuestros imposibles horarios de reuniones. Muchas gracias.

También agradecer todo el apoyo y amor recibido por Carlos y Antonia, mis padres, que desde pequeño han hecho posible que cumpla todos mis sueños.

Por último, deseo dar las gracias a Elena, la persona que me ha acompañado durante todo este viaje, desde el primer al último día. Ella me ha dado fuerzas para no rendirme nunca, ha sabido comprenderme cuando ni yo mismo lo hacía y ha sido el mayor motivo para seguir siempre adelante.

Víctor Díaz Yagüe

Me gustaría agradecer a todas las personas que han estado conmigo durante mi etapa universitaria, desde las fases más relajadas hasta las más duras, ya que con vuestra ayuda y apoyo he logrado alcanzar mis objetivos.

A mi familia, por estar conmigo durante este camino, Gracias.

A mis amigos, por todos esos consejos que me han sido útiles y los que no, por todo en general.

A todos mis profesores, porque gracias a vuestro esfuerzo, paciencia, dedicación y energía, nos habéis convertido en el futuro; llegar hasta aquí también es gracias a vosotros.

A mis compañeros del proyecto y prácticas, todas esas horas de trabajo duro por fin han tenido su recompensa.

Y a ti, no encuentro palabras para expresar lo mucho que te debo, porque ha sido tu fe y tu dedicación lo que ha hecho que me diera cuenta que podía empujar los suficiente para terminar esta etapa y conseguir lo que quería, gracias Irma.

Autorización

Los alumnos D. Juan José Arconada Alonso, D. Roberto Moreno Rojas y D. Víctor Díaz Yagüe; matriculados en el Grado de Ingeniería del Software impartido por la Facultad de Informática, autorizan a la Universidad Complutense de Madrid a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores el presente Trabajo de Fin de Grado, realizado durante el curso académico 2016 - 2017 bajo la dirección de D. Gonzalo Rubén Méndez Pozo y D^a. Raquel Hervás Ballesteros, profesores del Departamento de Ingeniería del Software e Inteligencia Artificial; así como a la Biblioteca de la Universidad Complutense de Madrid a depositarlo en el Archivo Institucional *E-Prints Complutense*, con el objetivo de divulgar la presente obra, promover la libre circulación del conocimiento y facilitar al público el acceso a contenidos culturales.

Madrid, 1 de Septiembre de 2017

Índice

Índice	i
Índice de figuras	iii
Lista de abreviaturas	v
Resumen.....	vii
Palabras clave.....	vii
Abstract	ix
Keywords.....	ix
1. Introducción.....	1
1.1 Motivación.....	2
1.2 Objetivos.....	3
1.3 Estructura de la memoria	3
2. Introduction	5
2.1 Motivations.....	6
2.2 Objetivos.....	7
2.3 Report structure	7
3. Trabajo relacionado	9
3.1 Realidad aumentada.....	9
3.2 Realidad aumentada enfocada en museos	11
3.3 Plataformas de realidad aumentada	13
3.4 Posicionamiento en el Museo	16
4. Tecnologías y recursos utilizados.....	17
4.1 Unity	17
4.2 Vuforia	18
4.2.1 Reconocimiento de imágenes en Vuforia.....	19
4.3 Museo García Santesmases.....	21
4.4 MongoDB.....	23
5. Funcionamiento de la aplicación	25
6. Diseño e implementación	31

6.1	Arquitectura del modelo de capas.....	31
6.2	Base de datos de la aplicación	34
6.2.1	Análisis y diseño de la base de datos	34
6.2.2	Atributos de la base de datos.....	36
6.2.3	Implementación de la base de datos	39
6.3	Diseño de la aplicación	42
6.3.1	Flujo de ejecución.....	43
7.	Conclusiones y trabajo futuro	51
7.1	Conclusiones	51
7.2	Trabajo futuro	52
8.	Conclusions and future work	55
8.1	Conclusions	55
8.2	Future work.....	56
9.	Aportaciones personales.....	59
9.1	Juan José Arconada Alonso	59
9.2	Roberto Moreno Rojas.....	61
9.3	Víctor Díaz Yagüe	63
	Manual de instalación	65
	Cliente.....	65
	Servidor.....	65
	Referencias	69

Índice de figuras

3.1 National Museum of Singapore, fotografía de Choo Yut Shing	12
4.1 Indicador Vuforia de calidad de imagen	19
4.2 Reconocimiento por NFT	20
4.3 Malla de puntos Vuforia	21
4.4 Pasillo del Museo García Santesmases	22
5.1 Interfaz inicial	25
5.2 Primera pista	26
5.3 Pista por comparación y pista de posición	27
5.4 Lupa desactivada y lupa activada	27
5.5 Control objeto visto	28
5.6 Objetos vistos y pistas obtenidas	29
5.7 Interfaz final	30
6.1 Arquitectura de capas	33
6.2 Esquema de arquitectura	33
6.3 Jerarquía de ordenación del museo	34
6.4 Elemento del museo	35
6.5 Estructura de datos inicial	37
6.6 Estructura de datos evolucionada	38
6.7 Estructura de datos final	39
6.8 Estadística Mongo DB Compass I	40
6.9 Estadística Mongo DB Compass II	41
6.10 Estadística Mongo DB Compass III	41
6.11 Estadística Mongo DB Compass IV	41
6.12 Diagrama de diseño	43
6.13 Flujo de ejecución	45
Transferencia de datos	65
Servicios Windows	67

Lista de abreviaturas

APK	-	<i>Android Application Package</i>
DAO	-	<i>Data Access Objects</i>
GPS	-	<i>Global Positioning System</i>
HUD	-	<i>Head up Display</i>
NFT	-	<i>Natural Feature Tracking</i>
PC	-	<i>Personal Computer</i>
RA	-	<i>Realidad Aumentada</i>
SaaS	-	<i>Software as a Service</i>
SDK	-	<i>Software Development Kit</i>
SQL	-	<i>Structured Query Language</i>
SLAM	-	<i>Simultaneous Localization And Mapping</i>
UCM	-	<i>Universidad Complutense de Madrid</i>
USB	-	<i>Universal Serial Bus</i>

Resumen

Es innegable que la sociedad se dirige hacia un futuro en el cual los computadores portátiles serán el centro del mismo, desde el desarrollo de la ciencia más compleja hasta el ocio más mundano.

La incorporación a la vida diaria de aparatos inteligentes es una realidad contrastable. Esto ha creado nuevos mercados y ofertas, empleándose los últimos avances tecnológicos para poder crear gadgets cada vez más complejos y a su vez dotarlos de nuevos sistemas software para trabajar con ellos.

La realidad aumentada es la técnica que permite integrar, empleando un dispositivo físico, una capa de software sobre el mundo que nos rodea, concediendo una interacción con la misma y consiguiendo “enriquecer” la experiencia del usuario con el entorno.

Este proyecto busca desarrollar un software de realidad aumentada que permita al usuario interactuar con el Museo García Santesmases de la Facultad de Informática de forma dinámica, otorgando una experiencia diferente.

La aplicación, en forma de juego, sirve para ofrecer al usuario nuevos datos y curiosidades no contenidas en el museo a la vez que le invita a explorarlo buscando un objeto aleatorio. Para ello se hace uso de un mecanismo de pistas que le llevarán desde un objeto inicial hasta el elemento final que se está buscando, haciendo uso de la cámara del soporte físico que se emplee. Mediante la comparación de las imágenes el programa será capaz de reconocer si el usuario ha logrado alcanzar el objeto buscado o si por el contrario se encuentra ante un elemento erróneo que le proporcionará una nueva pista de posición o de cualidad del objeto.

El proyecto se ha desarrollado empleando diferentes tecnologías. El software de Vuforia se ha utilizado como motor de realidad aumentada, Unity como motor gráfico sobre el que se despliega la información en el dispositivo móvil y MongoDB como base de datos para almacenar la información referente a los objetos.

Palabras clave

Interacción, museo, realidad aumentada, Unity, Vuforia.

Abstract

Undoubtedly, society advances toward a future where personal computer devices will be the center of it, from the hardest science development to the simplest leisure.

The incorporation into everyday life of more or less intelligent devices is a reality that can be contrasted. This has created new opportunities, markets and proposals, using the last technological advances to build increasingly complex gadgets, and at the same time endow them with new software to work with.

Augmented Reality is the technique that allows to integrate, using a physical device, a layer of software over the world that surround us, allowing an interaction with it, managing to “enrich” the user experience with the environment.

This project seeks to develop an Augmented Reality software that let the user to interact dinamically with García Santesmases Museum of the Facultad de Informática, granting him a different experience.

The application, in game form, serves to offer the user new data and curiosities not contained in the museum while inviting him to explore it by looking for a random object. This is done using a mechanism of tracks that will take him from an initial object to the final element that is being searched, making use of the camera of the physical medium that is used. By comparing the images, the program will be able to recognize if the user has achieved the desired object or if, on the contrary, he / she is confronted with an erroneous element that will provide a new track of position or attribute of the object.

The project has been developed using different technologies. Vuforia software has been used as an augmented reality engine, Unity as a graphics engine on which the information is displayed on the mobile device and MongoDB as a database for storing information concerning objects.

Keywords

Augmented reality, interaction, museum, Unity, Vuforia.

1. Introducción

En los últimos años se ha ido generando un interés creciente en la Realidad Aumentada (RA) y sus diversas aplicaciones en los diferentes campos de la industria.

Tiempo atrás se crearon aparatos “reactivos” para la lectura del entorno y la proyección de datos sobre él (los sistemas HUD de los aviones, por ejemplo) aunque si uno se ciñe a la definición más pura de ésta técnica (el empleo de tecnología para proyectar datos sobre el mundo real e interactuar con el mismo) notamos que el último punto no existía, carecíamos de la posibilidad de interacción.

La llegada del móvil inteligente así como de las tabletas digitales ha abierto un nuevo camino para todas estas técnicas, planteando oportunidades de integración con un mayor número de usuarios y abriendo nuevos mercados.

Paralelamente, los museos, como centros de conocimiento, necesitan estar en la vanguardia de los sistemas de divulgación, adaptándose a las tecnologías más atractivas tan pronto como sea posible. Hasta la aparición de estas herramientas los museos tenían limitada la interacción multimedia con sus visitantes, la mayoría de ellos recurrían a videos o dispositivos similares que no dejan de ser una suerte de exposición sobre la cual el usuario tiene poco control.

Con esto en mente, no han sido pocas las entidades públicas que han enfocado sus esfuerzos en emplear la realidad virtual o la RA como forma de mejorar la oferta que plantean. Grandes museos de todo el mundo, como el *Metropolitan Museum of Art*, el *Grant Museum of Zoology*, el *British Museum*, o el *Museo de América* ubicado en Madrid, han empezado a adaptar sistemas en torno a esta tecnología, buscando un punto diferencial que atraiga un mayor número de visitantes a base de ofrecer nuevas oportunidades respecto a cómo conectan, comparten y aprenden. Se trata de una forma de lograr una mayor participación de los visitantes y, de este modo, generar un mecanismo de integración que permita hacerles sentir parte del mismo, lo que se traduce en una mejor valoración del museo por parte de los clientes.

Además, otra cualidad a tener en cuenta es la posibilidad de volver a utilizar la aplicación y de que el museo pueda volver a ser visitado por un público que ya ha recorrido sus salas. Esto se debe a que la interacción puede ser diferente en cada visita, debido al dinamismo del software.

Atendiendo a estos valores, en el trabajo se plantea la opción de aplicar estas técnicas al Museo García Santesmases de la Facultad de Informática de la Universidad Complutense de Madrid, de forma que sus visitantes puedan disponer de un novedoso sistema que les permita conocer el museo desde otra perspectiva.

1.1 Motivación

El proyecto se selecciona entre los posibles candidatos porque permite experimentar con nuevas tecnologías dispares y novedosas, por ejemplo, algoritmos de posicionamiento, diferentes herramientas de RA y tecnologías móviles.

Desde hace unas décadas los elementos empleados por el ser humano, así como el entorno que lo rodea han evolucionado incrementando el uso de la tecnología en todos los ámbitos. El Museo García Santesmases, pese a poseer objetos tecnológicos importantes, carece de sistemas novedosos que faciliten su visita.

Por otro lado, a principios del verano del 2016, apareció en el mercado *Pokemon Go*. El juego causó un gran furor en la sociedad, que en su mayoría hizo uso por primera vez de la RA. El éxito causado sirvió para motivar los intereses en el desarrollo de este proyecto.

Todo ello nos motivó a construir una aplicación que sirviese de juego a la vez que permitiese visitar el museo. Para ello se hace uso de los dispositivos móviles que recogen información del entorno mediante la cámara, permitiendo posicionar a los visitantes en el museo además de ofrecerles información del entorno que les rodea. Así mismo, se establece la posibilidad de realizar diferentes recorridos de forma aleatoria, interrelacionando los objetos del mismo y aportando datos curiosos.

A pesar de que todo el software se encuentra enfocado al Museo García Santesmases, la estructura de datos elegida, definida más adelante, así como la

mecánica de los algoritmos, permite la ampliación del museo y la actualización de la aplicación sin grandes dificultades, cabiendo incluso la posibilidad de adaptarse a otros recintos con una distribución similar.

1.2 Objetivos

A continuación, se describen los objetivos que se han planteado a lo largo de este proyecto:

- Generar una aplicación móvil multiplataforma accesible para cualquier dispositivo *Android* o *iOS*.
- Interactuar con los objetos del museo mediante el reconocimiento de imágenes.
- Posicionar al usuario en función de los elementos que enfoque con su dispositivo móvil para establecer el camino que ha recorrido.
- Documentar los elementos del museo para poder relacionarlos entre sí, posibilitando un mecanismo de juego y mostrando la información al usuario de manera atractiva.
- Establecer un formato de juego que resulte interesante utilizando la información recabada de los objetos del museo.
- Crear una interfaz intuitiva para que el jugador pueda interactuar con la aplicación de forma sencilla y amigable.

1.3 Estructura de la memoria

La memoria está distribuida en nueve capítulos que describen el progreso del proyecto desde su inicio. En ellos se detalla el desarrollo de la idea inicial y el proceso de evolución llevado a cabo.

En el primer y segundo capítulo se realiza una introducción al proyecto. Se describe la motivación del mismo, los objetivos, así como la descripción de la estructura de la memoria. El segundo está traducido al inglés.

En el tercer capítulo se comenta el estado del arte, referido a la realidad aumentada enfocada en museos y sus plataformas. Del mismo modo, se describe el trabajo ya existente de posicionamiento para el Museo García Santesmases.

Es importante destacar que este capítulo se inicia con una introducción a la realidad aumentada para establecer una serie de conceptos fundamentales y se describe cómo se están adaptando otros museos a esta técnica.

En el cuarto capítulo se detallan todas las tecnologías y recursos utilizados para el desarrollo de la aplicación.

El quinto capítulo contiene una descripción del funcionamiento del programa mostrando los estados de la interfaz gráfica en todas sus fases.

En el sexto capítulo se describe la arquitectura de la aplicación por capas, además de detallar la relación entre el servidor y el cliente que componen el esquema general del proyecto. También se especifica el proceso de análisis, diseño e implementación de la base datos. Por último, se describe el diseño del software mediante el flujo de ejecución.

Los capítulos séptimo y octavo comprenden las conclusiones y el trabajo futuro del proyecto. El primero en castellano y el segundo en inglés.

En el siguiente capítulo, número nueve, se relatan las aportaciones personales de cada uno de los miembros del equipo.

Al final del documento se encuentra un apéndice que contiene el manual de instalación de la aplicación.

2. Introduction

During the last years there has been a growing interest in Augmented Reality and its various applications in different industry fields.

Some time ago, reactive devices were created for reading environment and projecting data on it (aircraft HUD systems is a good example), although if one adhere himself to the purest definition of this technique (the use of technology to project data about real world and interact with it) it's easily identifiable that the last point did not exist: we lacked the possibility of interaction.

Arrival of Smart mobiles as well as digital tables has opened a new path for all these techniques, posing opportunities for integration with a greater number of users, and creating new markets.

At the same time, museums, as centers of knowledge, need to be at the forefront of outreach systems, adapting the most attractive as soon as possible. Until the appearance of these tools, museums had limited multimedia interactions of their visitors to videos or similar devices, which are a kind of exhibition over which the user has zero to none control.

Taking this into account, there have been a few public entities that have focused their efforts on using Virtual Reality or Augmented Reality as a way to improve their offer. Large museums, such as the Metropolitan Museum of Art, Grant Museum or British Museum, in other countries, and even some spanish ones like Museum of Americas, have begun to adapt systems around this technology, looking to create a differential point that attract more visitors by offering new opportunities for how they connect, share and learn. It is a way to achieve a greater participation of these people and generate, in this way, an integration mechanism that allows them to feel part of it, which translates into a greater and better valuation of the museum.

In addition, another quality to be taken into account is the possibility of reusing the application and that the museum can be visited again by an audience that has already visited all its rooms. This is because the interaction can be different in each visit, due to the dynamism of the software.

Keeping this in mind, the options of giving a similar life through these techniques to the García Santesmases Museum of the Faculty of Computer Engineering of Universidad Complutense de Madrid is proposed.

2.1 Motivations

The project is selected among potential candidates because it allows us to experiment with new and diverse technologies, for example, positioning algorithms, different RA tools and mobile technologies.

Since some decades ago the elements used by human beings, as well as its surrounding environment, have evolved increasing the use of technology in all areas. García Santesmases Museum, despite possessing important technological objects, lacks innovative systems that facilitate its visit.

On the other hand, during early 2016 summer appeared in the market Pokemon Go. The game caused a great impact in society, which made use of the RA for the first time in a massive way. This success generated new interests in the development of this project.

All of this motivated us to build an application that served as a game while allowing visitors to visit the museum. To achieve this mobile devices that collect information from the environment through the camera are used, allowing visitors to be positioned in the museum while offering information about the environment around them. In addition, it establishes the possibility of making different routes in a random way, interrelating objects alike and providing curious data.

Although all the software is focused on the García Santesmases Museum, the chosen data structure, defined later, as well as the mechanics of the algorithms, allows the expansion of the museum and the update of the application without great difficulties, including the possibility of adapting it to other venues with a similar distribution.

2.2 Objectives

The following are the objectives that have been proposed throughout this project:

- Achieve interaction with museum objects through image recognition.
- To document the elements of the museum in order to be able to relate them one to another, enabling a game mechanism and displaying information to the user in an attractive way.
- To create an intuitive interface so that the player can interact with the application in a simple and friendly way.
- To position the user, using the elements that are focused with his mobile device to establish the path that has traveled.
- To establish an interesting game format using information collected from museum objects.
- To generate an accessible cross-platform mobile application for any kind of device, Android or iOS alike.

2.3 Report structure

This document is distributed in nine chapters that describe the progress of the project since its inception. It details the development of the initial idea and the process of evolution carried out.

First and second chapter are an introduction of the project. It describes the motivation of it, the objectives, as well as the description of the structure of this report. The second is translated into English.

The third chapter discusses the state of art, referred to the augmented reality focused on museums and their platforms. Similarly, the existing positioning projects for the García Santesmases Museum are described.

It is important to note that this chapter begins with an introduction to augmented reality, to establish a series of fundamental concepts, and describes how other museums are adapting to this technique.

Fourth chapter details all the technologies and resources used to develop the application.

Fifth chapter contains a visual description of the software, showing the states of the graphical interface in all its phases.

The sixth chapter describes the architecture of the application by layers, in addition to detailing the connection between the server and the client that compose the general scheme of the project. It also specifies the process of analysis, design and implementation of the database. Finally, the design of the software is described through the execution flow.

Seventh and eighth chapters comprise the conclusions and future lines of the project. First one in Spanish and second one in English.

Next chapter, number nine, the personal contributions of each of the team members are reported.

At the end of the document there is an appendix containing the application installation manual.

3. Trabajo relacionado

En este capítulo se introducen algunos conceptos fundamentales que se requieren para la comprensión del proyecto, así como una visión general sobre la evolución de la tecnología que se utiliza y el estado de la misma según se emplea en entornos similares.

3.1 Realidad aumentada

“La realidad aumentada (RA) [1] es el término que se usa para definir la visión de un entorno físico del mundo real, a través de un dispositivo tecnológico, es decir, los elementos físicos tangibles se combinan con elementos virtuales, logrando de esta manera crear una *realidad aumentada* en tiempo real.”

Se ha empleado como referencia para obtener una mayor información sobre RA el libro “Augmented Reality: An Emerging Technologies Guide to AR”. [2]

Sintetizando la información obtenida de las fuentes anteriores se extrapola la siguiente definición. La realidad aumentada es el empleo de dispositivos tecnológicos, generalmente móviles, para capturar el entorno que nos rodea y combinarlo con elementos digitales, ampliando en tiempo real la información que obtenemos de ellos y permitiéndonos interactuar de una forma intangible con ella.

La RA se diferencia de la realidad virtual en que es necesario un entorno físico real sobre el que desplegar todos estos elementos digitales para llevarla a cabo. No se recrea un entorno virtual sobre el que desplazarse, sino que es la misma realidad de la que se extrae la información necesaria.

Esto permite una mayor integración e inmersión, ya que no necesitamos aislarnos del lugar en el que nos encontramos, lo que nos deja pensar en otras posibilidades que la realidad virtual no puede ofrecer.

Esta tecnología se concibió de forma básica en la década de 1960, aunque es en la actualidad, cuando los avances técnicos han permitido el acceso ilimitado a una red de datos fluida, que ha posibilitado junto con la creación de dispositivos

portátiles con el suficiente potencial y las capacidades de procesamiento necesarias el desarrollo exponencial de la RA.

Sin embargo, a medida que los usuarios descubrían su potencial los desarrolladores comenzaban a idear nuevas formas de usos, educativos, de entretenimiento, militares o legales, por citar algunos, generando que nuevos mercados explotasen esta tecnología. Lo que comenzó siendo una herramienta de mercadotecnia se ha convertido en algo fundamental en el día a día, siendo utilizado en muchas situaciones sin ni siquiera ser conscientes de ello, por ejemplo, cuando vemos un partido de algún deporte y se sobrepone a la imagen un gráfico indicativo de alguna característica, en ese caso se está experimentando la RA.

De igual manera que la realidad virtual tiene diversos campos y exige diferentes métodos, la RA se ha adaptado en función de las necesidades del mercado.

Greg Kipper en su libro, referenciado al comienzo de este apartado, cita al menos cuatro plataformas diferentes:

1. Computadoras personales con webcams

El hecho de que un PC disponga de todos los componentes necesarios para emplear la RA hace que su adaptación sea inmediata. Se pueden encontrar ejemplos de RA en las aplicaciones que acompañan a las cámaras web, y que permiten insertar gráficos sobre las imágenes capturadas.

2. Kioskos digitales y displays

Interactúan con una imagen capturada y permiten la inserción de datos específicos. Un ejemplo de ello son los que se suelen encontrar en las exposiciones de grandes ferias.

3. Smartphones y tabletas

Representan el paradigma hacia el cual se ha enfocado con más empeño todo el desarrollo de la RA. El que casi todos ellos incluyan hoy en día un sistema de captura de imágenes y de posicionamiento los convierte en plataformas ideales para ello, tomando la posición del usuario y generando datos en pantalla en función de la misma.

4. Gafas RA y dispositivos de cabeza

Por el momento, son el nicho de mercado más pequeño. Tal vez sea demasiado temprano para ellos, aunque podemos encontrar ejemplos de su uso en los cuerpos militares y de seguridad.

Todos estos aparatos dependen del software que se despliegue en ellos. El principal reto en los años venideros es conseguir programas capaces de trabajar con el tremendo flujo de información al que se puede acceder y mostrarlo de forma coherente en el dispositivo empleado. Este material, íntegramente, requerirá a su vez importantes plataformas hardware para llevar a cabo el procesamiento de estos datos y conexiones móviles de altísima calidad, colocando a la RA como la punta de lanza de varios campos de desarrollo, permitiendo interesantes oportunidades.

3.2 Realidad aumentada enfocada en museos

Fuera del entorno de los museos existen multitud de aplicaciones que emplean esta técnica, siendo la más exitosa *Pokemon Go*, que revolucionó el mercado durante el 2016, en la cual el usuario se desplaza por el entorno que le rodea, empleando la geolocalización GPS del dispositivo para ubicarse dentro del mundo virtual, “cazando” *pokemons* que tan solo pueden visualizarse empleando la cámara de su aparato móvil y el propio software. Esta información ha sido obtenida de las siguientes referencias [3], [4], [5] y [6].

Si uno se ciñe al aspecto académico se puede hablar de varias entidades que han hecho de la realidad aumentada su máximo estandarte.

El *Jinsha Site Museum* [7] [8] [9], en China, emplea la RA para recrear los elementos empleados por las civilizaciones que habitaron hace tres mil años el lugar en el que está emplazado, creando un entorno 3D en los dispositivos móviles, réplicas de las reliquias y los artefactos allí encontrados. La interacción del mismo se ciñe a descubrir cosas sobre ellos y obtener imágenes más o menos realistas de los mismos.

Un paso más allá va el *Kennedy Space Center* [10], ya que a la generación de elementos en 3D y a la muestra de información en el hardware empleado para interactuar con el entorno, en algunos sitios pantallas de vidrio y en otros

dispositivos móviles, se une la recreación de hologramas de héroes y astronautas que formaron parte de los descubrimientos espaciales. Estos hologramas “hablan” con el usuario, revelándole curiosidades y contándole aspectos de su profesión o simplemente reinterpretan eventos acontecidos como una suerte de video proyectado sobre el entorno real.

El *Smithsonian Institution* [11] [12] emplea la RA en una de sus salas para recrear, sobre los huesos expuestos en una de sus exposiciones de biología, el aspecto externo de los animales a los que pertenecen. De esta forma le otorga “vida” a la exposición, permitiendo que el usuario aprecie cómo se movían o incluso cómo cazaban los mismos. Esta institución emplea una aplicación que está disponible al público sin necesidad de acudir al lugar.



3.1 National Museum of Singapore, fotografía de Choo Yut Shing

Sin embargo, quizás el más completo sea el *National Museum of Singapore* [13] [14], que permite a los visitantes jugar dentro del mismo, de una manera muy similar a lo que permite *Pokemon Go*, teniendo que localizar una serie de plantas en el museo empleando las cámaras de sus móviles. Esta aplicación es muy similar a la que finalmente se ha desarrollado como proyecto, y el éxito de este museo nos permite ser optimistas a la hora de pensar en diferentes salidas para nuestra aplicación.

En nuestro país podemos encontrar ejemplos similares como el *Museo de América* en Madrid, en el cual se desplegó un proyecto para dotar de vida un mapa mudo con el que los visitantes puedan interactuar para recibir información de lo que están observando. Este trabajo [15] fue llevado a cabo por miembros y alumnos de la Facultad de Informática de la Universidad Complutense.

Otro ejemplo es la aplicación creada para el *Museo Thyssen-Bornemisza* [16], "Crononautas, un viaje en el tiempo en el Museo Thyssen". Esta aplicación, muy similar a la planteada inicialmente en este proyecto, genera un hilo argumental inicial para a continuación mostrar diversas alternativas al usuario, ofreciéndole la posibilidad de interactuar con las obras del museo.

3.3 Plataformas de realidad aumentada

Existe una cantidad numerosa de plataformas de RA en el mercado, tanto *Open Source* como de código propietario, enfocadas a diferentes tareas en función del objetivo para el que se han creado.

De entre todas ellas cabe destacar algunas, las consideradas más avanzadas, así como las que se plantearon como posibles candidatas para emplearlas en el desarrollo del proyecto, como son *ARToolKit*, *Augment*, *ATOMIC*, *NyArToolKit*, *Kudan*, *EasyAR*, *Wikitude* o *Vuforia*, la cual recibe un apartado propio más adelante.

En la página web SocialCompare [17] puede verse una comparativa de las diferentes características de las plataformas de RA y, en algunos casos, los motivos que causaron que se descartase su uso.

ARToolKit

Es una librería de seguimiento, *Open Source*, que permite la generación de contenidos de RA sobre imágenes reales [18] [19]. Es una de las más empleadas para estas tareas. Durante los dieciocho años que ha sido desarrollada ha permitido resolver con éxito dos de los problemas fundamentales que existen en la RA, como son el seguimiento de un punto del entorno y la interacción con el mismo.

Está preparada para funcionar tanto en *iOS* como en *Android*, así como *Windows*, *OS X* o *Linux*.

A pesar de ser una de las opciones que se contemplaron para el desarrollo del proyecto, el hecho de que esta librería sea tan popular y con tantas funcionalidades diferentes dificultaba su aprendizaje y complicaba profundamente su uso por lo que finalmente fue descartada.

NyARToolKit

NyARToolKit [20] es una versión simplificada de ARToolKit que contiene unas funcionalidades reducidas pero válidas para trabajar con ellas en este proyecto. En vista a los problemas que daba la ingente cantidad de funciones de ARToolKit se optó por probar esta librería, pero fue rechazada debido a que no existía una versión que se integrase en inglés, ya que todo el desarrollo de la misma procede de Japón.

ATOMIC

Herramienta de desarrollo creada como una interfaz gráfica para el uso de ARToolKit.

Es multiplataforma y, frente a todas las demás aquí presentadas, ATOMIC [21] busca facilitar la creación de aplicaciones de RA a través de una interfaz amigable sin necesidad de tener amplios conocimientos técnicos de programación.

Augment

Augment [22] [23], inicialmente un software *Open Source*, es una plataforma para desarrollo de RA que se basa en el paradigma SaaS. Se enfoca hacia el e-commerce (con la generación de gráficos integrados en imágenes reales a tiempo real) y puede ser empleada tanto en aplicaciones móviles como en desarrollo web. Emplea parte del SDK de otra herramienta de trabajo en RA, Vuforia.

Se despreció su uso debido a los costes de trabajar con la herramienta y a las limitaciones que presentaba su carácter comercial.

EasyAR

Motor de desarrollo de RA, preparado para funcionar en las principales plataformas móviles (*Android*, *iOS*, *UWP*) así como en *Windows* y en *OS X*. Actualmente en la versión de la herramienta 2.0, que permite reconocimiento 3D y trabajo con paquetes en la nube. Está considerada una de las mejores opciones del mercado, especialmente porque su empleo es libre. [24]

Se descartó su uso ya que cuando se comenzó este proyecto sólo existía la versión 1.3, que presentaba limitaciones a la hora de reconocer imágenes volumétricas.

Kudan

Es otra de las mejores opciones del mercado para el desarrollo de trabajos en RA [25] [26]. Multiplataforma, con capacidad de desarrollo *cross-platform* a través de Unity3D, emplea la tecnología SLAM (tecnología que permite posicionar un objeto dentro de un mapa que se está creando en ese mismo momento) para potenciar sus capacidades de geolocalización.

Para la base de datos de objetos de reconocimiento se puede emplear el editor de Unity, lo que por una parte es interesante ya que simplifica el proceso, pero por otro es un inconveniente ya que es propenso a errores.

El alto coste de una licencia no limitada y ciertas dificultades de integración hicieron que se desechase su uso.

Wikitude

Wikitude [27] [28] está considerada la mejor herramienta de generación de contenido para RA. Permite el seguimiento de objetos empleando SLAM, seguimiento de imágenes 3D, trabajo con datos geolocalizados y reconocimiento de datos en la nube, entre otras características, así como capacidades nativas tanto en *Android* como en *iOS* y una SDK tanto en *Java* como en *Javascript* que facilita su desarrollo, lo que lo convierten en un candidato idóneo para el empleo en la creación de cualquier aplicación.

El problema principal radica en su alto coste de licencia por aplicación desarrollada.

Vuforia

Vuforia [29] [30] es una plataforma de desarrollo de software para RA. Sus características permiten trabajar con un motor de reconocimiento de imágenes, con los sensores de posición del aparato empleado y la cámara del mismo, uniendo todo ello y permitiendo desplegar sobre la imagen capturada un nuevo entorno virtual, coherente con la realidad. Además, es multiplataforma lo que permite realizar desarrollos para *Android* o *iOS*.

Este ha sido el motor de desarrollo de RA elegido para el proyecto porque se ajusta a las necesidades del mismo.

3.4 Posicionamiento en el Museo

Debido a la necesidad de posicionar al usuario en el Museo García Santesmases, se contempló la posibilidad de utilizar un algoritmo desarrollado para tal fin en proyectos anteriores.

Los alumnos de la Facultad de Informática, Christian Álvarez y Kevin Arboleda, [31] emplearon el posicionamiento con una técnica de triangulación de señales WiFi. Para llevar a cabo esta técnica desarrollaron una evolución del software previamente implementado por Enrique López, Francisco Javier Moreno y Javier Plá [32]; también alumnos de dicha facultad, que ya permitía establecer un objeto en una posición determinada de los pasillos de la Universidad.

Se descartó emplear esta técnica porque existía un conflicto entre la tecnología de Vuforia y la plataforma de desarrollo elegida. Se optó por realizar un método de posicionamiento basado en el reconocimiento de imágenes de Vuforia.

4. Tecnologías y recursos utilizados

En este capítulo se describen las diferentes tecnologías utilizadas para el desarrollo de la aplicación, así como los recursos empleados para posibilitar el correcto funcionamiento del software.

4.1 Unity

Unity [33] [34] es un motor de juegos multiplataforma, desarrollado en torno al 2004, compatible con los más importantes programas de diseño y modelado gráfico, facilitando la integración de recursos creados por estos últimos. Emplea las APIs para producir gráficos OpenGL, Direct3D, OpenGL ES o algunas propietarias (en el caso del desarrollo para Wii)

Para poder trabajar con Unity se ha empleado el lenguaje C#, aunque también se puede emplear un lenguaje derivado de *JavaScript* y *Boo*.

Además de la integración con Vuforia existen otros motivos que facilitaban la elección de Unity como plataforma de desarrollo.

Uno de los mayores beneficios, especialmente a la hora de empezar a trabajar con él, es que gestiona los recursos del sistema que emplea por sí mismo, lo que permite al desarrollador centrarse más en la funcionalidad y obviar ese apartado, esquivando de paso una previsible cantidad de bugs, aunque cediendo una pequeña parte de control a cambio.

Además, detrás de todo el desarrollo de Unity existe una comunidad muy numerosa que genera librerías públicas para ser usadas en la creación de juegos propios. A pesar de que muchas de ellas son de pago, existen suficientes recursos gratuitos como para generar casi cualquier proyecto desde cero.

Por otra parte, el coste de una licencia de Unity depende del plan que se elija, siendo cero para desarrollos personales y de \$35 o \$125 al mes en el caso de los planes profesionales. Para este proyecto se ha empleado la licencia gratuita ya que cubre todas las necesidades del mismo.

4.2 Vuforia

Como se ha mencionado al final del anterior capítulo, Vuforia ha sido la plataforma de desarrollo software elegida para la utilización de la RA en el proyecto. El hecho de que sea multiplataforma permite cumplir con uno de los objetivos planteados.

Vuforia dispone de algoritmos de reconocimiento de imágenes 3D para buscar en tiempo real objetos con formas determinadas (esferas, cajas) o bien representaciones y modelos específicos. Esto posibilita que se pueda crear software empleando Vuforia que busque en el entorno elementos determinados sobre los que desplegar una “capa de pintura” de información, o nuevas imágenes virtuales sobre los mismos.

Vuforia analiza constantemente el entorno, lo que permite que las imágenes desplegadas se actualicen en tiempo real y adopten la posición adecuada respecto al usuario, su posición y el objeto sobre el cual se muestra, dando una apariencia real.

De entre todas las opciones que se barajaron se eligió Vuforia por su integración sencilla con Unity y las opciones de reconocimiento de imágenes que ofrecía. Además, la versión de desarrollo no está limitada a una cifra de operaciones baja que complicase el trabajo con la misma.

Por otra parte, la posibilidad de generar los propios puntos de reconocimiento a partir de objetos del entorno, empleando *Vuforia Object Scanner*, que permite registrar el entorno y crear *targets* sobre los que trabajar y almacenar opcionalmente en la nube o bien en local, aportaba una flexibilidad que no entregaban las demás herramientas probadas.

En noviembre del 2016 Vuforia y Unity se asociaron, lo que les llevaría a integrar la primera en el paquete de desarrollo del segundo a lo largo del año 2017. Esto hace pensar que en un futuro el trabajo con ambas tecnologías será aún más sencillo.

4.2.1 Reconocimiento de imágenes en Vuforia

Para describir el funcionamiento del reconocimiento por imágenes de Vuforia [35] conviene explicar cómo se almacenan y cuáles son los pasos a llevar a cabo para poder trabajar con ello.

Una de las opciones de Vuforia es el uso de un *dataset* (conjunto de datos) generado en la nube. Para poder emplearlo es necesario registrar una licencia, a la que será asociado un identificador. Ese identificador determina de manera unívoca el *dataset* elegido y deberá incluirse en el código de la aplicación para poder acceder a él. Todo ello se lleva a cabo desde la página web de Vuforia haciendo uso de su sistema de almacenamiento.

En este *dataset* en la nube se incluirán los ficheros desde los que se extraerán los datos que empleará Vuforia para reconocer las imágenes. Para poder identificar las imágenes mostradas Vuforia genera una malla de puntos, extraídos de los detalles más relevantes de las fotos aportadas al mismo, por lo que una de las condiciones de las imágenes es que contengan los suficientes puntos de contraste como para generar una red práctica. Este paso está controlado por el servidor de Vuforia, indicando al desarrollador si la fotografía es válida o si, por el contrario, es demasiado pobre para poder emplearse. Para ello utiliza un sistema de puntuación basado en estrellas.

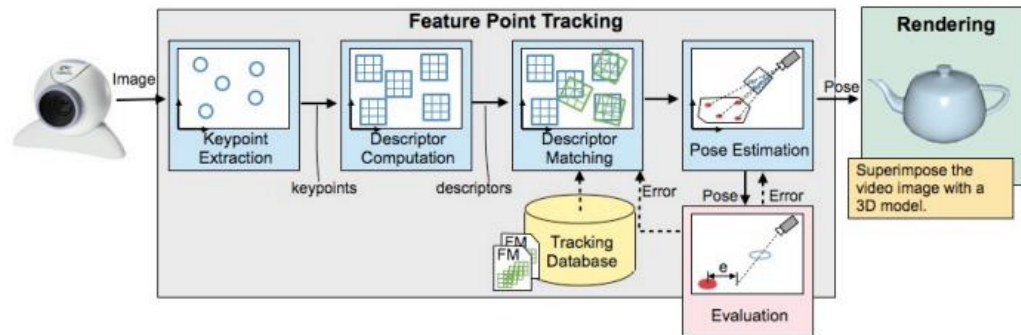
<input type="checkbox"/> Target Name	Type	Rating	Status
<input type="checkbox"/>  stones_hires	Single Image	★★★★★	Active

4.1 Indicador Vuforia de calidad de imagen

Una vez creada la malla se debe descargar y almacenar en el dispositivo sobre el que va a ejecutarse la aplicación, o bien se puede conservar en la nube para ser distribuida a petición. Es esa red de puntos la que hará que Vuforia reconozca los objetos que tenga delante.

Este sistema de identificación se denomina *Natural Feature Tracking* [36] y permite reconocer imágenes a partir de un subconjunto de puntos que se sabe que existen en un conjunto dado. Esta técnica, si bien plantea ciertos problemas para las imágenes demasiado similares, es la más empleada actualmente para la

generación de entornos de RA y prácticamente todos los motores de desarrollo se apoyan en ella.



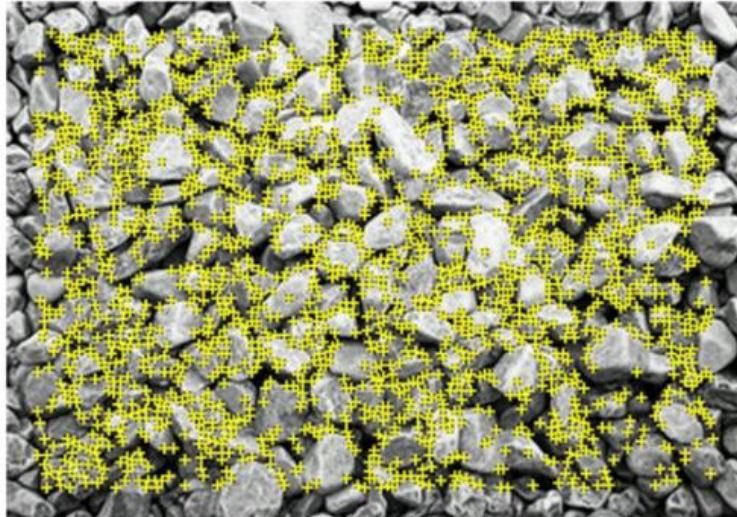
4.2 Reconocimiento por NFT

El proceso interno que sigue Vuforia para llevar a cabo la corrección de los problemas asociados a esta técnica y la identificación de los objetos queda fuera del alcance, ya que el código de Vuforia no es *Open Source* y no permite su análisis detallado.

En la siguiente imagen podemos apreciar la malla de puntos generada mediante el sistema *Vuforia Object Scanner*.

stones_hires

[Edit Name](#) [Remove](#)



4.3 Malla de puntos Vuforia

4.3 Museo García Santesmases

El Museo García Santesmases se encuentra situado en la tercera y cuarta planta de la Facultad de Informática de la Universidad Complutense de Madrid, bajo la tutela de su director, D. José Manuel Mendías Cuadros.

En él se recogen piezas fundamentales de la informática en España, dispositivos empleados por diferentes departamentos de la UCM, así como objetos donados por particulares que tuvieron más o menos relevancia en el contexto de la historia de la Ingeniería Informática.



4.4 Pasillo del Museo García Santesmases

El museo está distribuido en dos plantas, ocupando los pasillos laterales de ambos pisos. En ellas se pueden encontrar elementos de gran volumen (analizadores diferenciales, computadores analógicos, paneles de control de industria) ocupando segmentos enteros, así como elementos más pequeños (procesadores, sistemas de almacenamiento de datos, consolas), agrupados en vitrinas.

Todos los objetos están expuestos en espacios propios protegidos por una mampara cristal y organizados en base a su funcionalidad. Se encuentran brevemente documentados con su nombre, año de fabricación y en algunos casos una breve descripción.

El Museo García Santesmases se inauguró en el mes de noviembre del 2003 y desde entonces ha ido incrementando el número de elementos de la colección gracias a las donaciones de particulares y a la adquisición de nuevos objetos.

Además de las piezas contenidas en él, se puede encontrar información al respecto en la biblioteca de dicha facultad.

El museo cuenta con una web propia que amplía los datos allí ofrecidos. Sin embargo, a la fecha de redacción de esta memoria, la web no ofrece la totalidad

de sus funcionalidades. Por ejemplo, la visita virtual no está disponible para ninguno de los navegadores probados.

Los datos existentes del museo al comienzo del proyecto eran insuficientes para la realización del mismo. Por ello fue necesario realizar un importante trabajo de documentación.

4.4 MongoDB

MongoDB [37] es una base de datos no relacional que permite almacenar información de forma sencilla utilizando el formato JSON. Esto favorece la modificación del esquema utilizado para la creación de la estructura de datos cuando las aplicaciones evolucionan. Además, existen las diversas funcionalidades que una base de datos convencional posee como la utilización de índices o las búsquedas complejas.

MongoDB ayuda al aumento del rendimiento en las lecturas y escrituras de información al emplear la computación en memoria y ofrece estabilidad al implantarlo en servidores con grandes estructuras complejas de datos.

Presentan la posibilidad de contratar un servicio de asistencia profesional para licencias comerciales y acceso a recursos especiales, entre los que se encuentran, aumentar la seguridad, facilitar las posibles evoluciones de la base de datos, monitorización y respaldo en la nube. Aparte de las licencias, MongoDB ofrece un servicio de consultoría para ayudar a los clientes, por ejemplo, implantar la aplicación en cualquier entorno de desarrollo creando un nuevo sistema desde su inicio o migrando desde otras bases de datos.

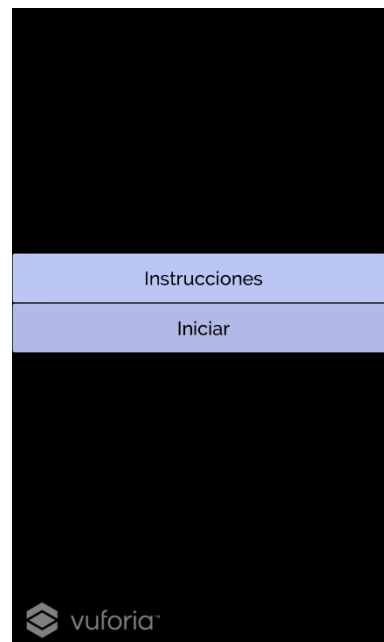
Cabe destacar que ofrecen herramientas gratuitas como un entorno de pruebas con funcionalidades básicas de la herramienta o una aplicación de gestión de la base de datos denominada *MongoDB Compass*, que nos permite visualizar y explorar los datos además de insertarlos, modificarlos o eliminarlos, e incluso opciones de depuración y optimización.

Para el desarrollo de este proyecto se eligió esta herramienta porque facilita el tratamiento de los datos, es sencilla de implementar y permite almacenar la información sin una estructura previamente acordada.

5. Funcionamiento de la aplicación

En este capítulo se detalla el completo funcionamiento de la aplicación desarrollada en este proyecto. Se trata de un juego que sirve para conocer el museo García Santesmases apoyándose en la búsqueda de un objeto aleatorio con la ayuda de pistas, que se van obteniendo al enfocar otros objetos con la cámara del dispositivo móvil.

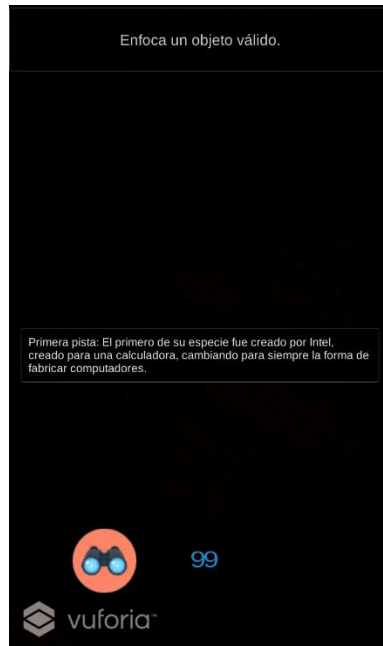
Al abrir la aplicación aparece la siguiente pantalla donde el usuario podrá elegir entre observar las instrucciones e iniciar el juego.



5.1 Interfaz inicial

Si se selecciona la opción “Instrucciones” se mostrará una breve descripción con la mecánica del juego para que el usuario comprenda el funcionamiento del mismo y de todos los elementos que dispone.

Al seleccionar “Iniciar” el juego se pondrá en marcha. El usuario visualizará en la pantalla una curiosidad del objeto que debe encontrar a modo de pista. En la siguiente imagen podemos observar este hecho, además en la parte superior de la pantalla se le indica al jugador que debe enfocar un objeto valido para que continúe recibiendo pistas.



5.2 Primera pista

A partir de aquí el usuario debe ir enfocando elementos del museo. Por cada elemento que visualice con el dispositivo móvil se le mostrará una pista. Esta pista puede ser de dos tipos, de comparación o de posición.

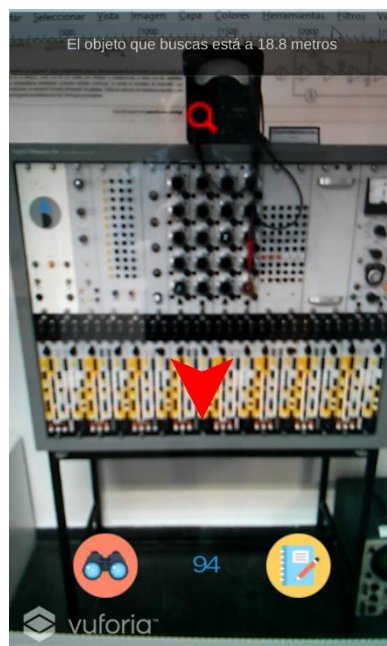
Primero se cotejan los atributos del objeto enfocado en relación con el elemento buscado y en caso de que existan coincidencias se le muestra al usuario una pista por comparación. Si no hay ninguna coincidencia entre los objetos se le muestra al usuario la pista de posición.

Las pistas por comparación revelan una cualidad del objeto que se está buscando y que comparte con el elemento del museo enfocado en ese momento. Las de posición muestran la distancia en metros que hay desde el elemento enfocado hasta el objeto buscado. Además, se muestra una flecha que indica la dirección en la que se encuentra el objeto que se desea localizar. En las siguientes imágenes se puede observar un ejemplo de cada una de las pistas, en la parte superior de la pantalla se visualiza como se indica al jugador que este no es el objeto que busca.



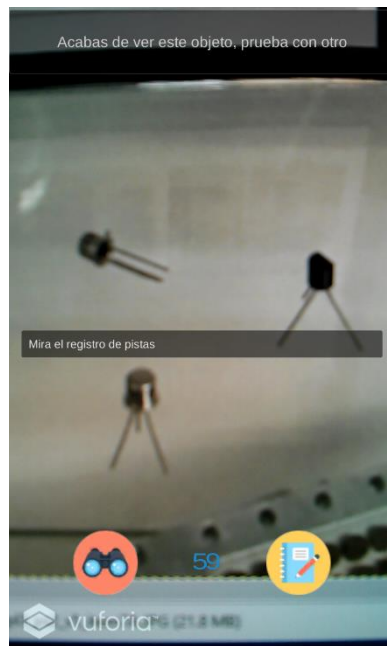
5.3 Pista por comparación y pista de posición

Cuando el usuario visualiza un objeto con el dispositivo móvil le aparece una lupa de color rojo en la parte superior de la pantalla que le permite conocer el nombre y la descripción del objeto que está viendo.



5.4 Lupa desactivada y lupa activada

Si el jugador enfoca un objeto por segunda vez le aparecerá un mensaje en la parte superior, en el que se indica que ese objeto ya ha sido visto. Además, en lugar de volver a darle la pista relacionada con ese objeto se le mostrará otro mensaje en mitad de la pantalla en el que se señala que mire el registro de pistas.

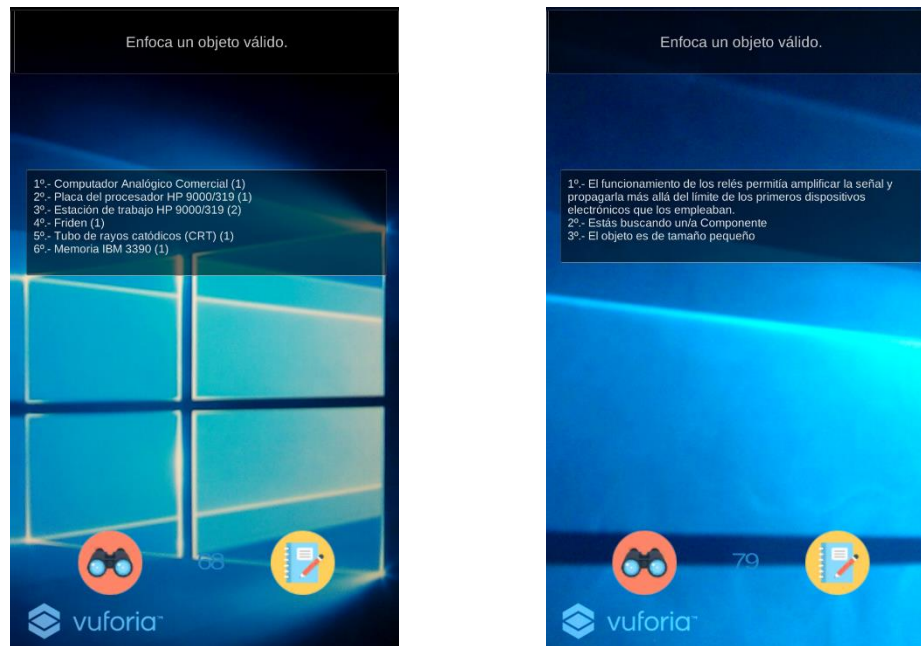


5.5 Control objeto visto

El usuario también dispone de dos botones. El primero de ellos está situado en la parte inferior izquierda, es de color rojo anaranjado y tiene el dibujo de unos prismáticos en el centro. Su función es mostrar el registro de objetos enfocados con el móvil, al pulsarlo aparece una lista con los nombres de los objetos visualizados y entre paréntesis el número de veces que se ha visto ese elemento.

El segundo botón se encuentra en la parte inferior derecha, es de color amarillo y tiene el dibujo de un cuaderno con un lápiz. La función de este botón es mostrar el registro de pistas que ha obtenido el usuario enfocando elementos del museo, al presionar este botón aparece una lista con las pistas.

A continuación, se pueden observar dos imágenes con el aspecto que tiene la interfaz gráfica al pulsar estos botones.



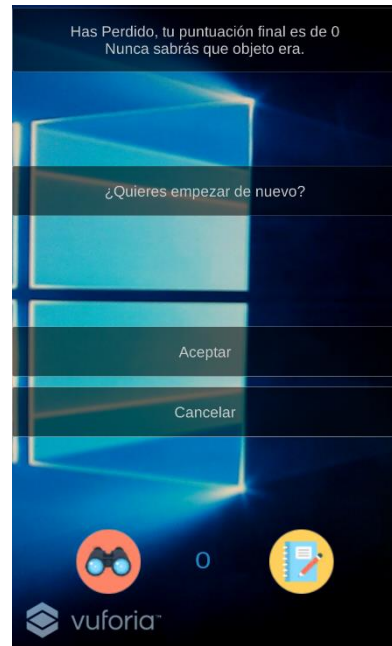
5.6 Objetos vistos y pistas obtenidas

El objetivo del juego, como se ha mencionado al comienzo de este capítulo, es encontrar un objeto aleatorio del museo, pero además existe una puntuación, representada en color azul en la parte inferior de la pantalla, que se decrementa cada vez que se enfoca un elemento que no es el objeto final. Si esta puntuación llega a cero el jugador habrá perdido la partida y no sabrá cuál es el objeto que estaba buscando. La idea es encontrar el elemento buscado obteniendo la máxima puntuación posible.

En caso de ganar se mostrará un mensaje en la parte superior del móvil indicando tal hecho y a continuación la puntuación obtenida en la partida. En el centro de la pantalla se le preguntará al usuario si quiere volver a jugar y aparecerán justo a continuación dos opciones: Aceptar o Cancelar.

En el supuesto de perder se indicará mediante un mensaje en la parte superior del móvil, de igual modo que cuando se gana se le preguntará al usuario si quiere volver a jugar.

En las siguientes imágenes se puede observar el estado de la interfaz gráfica en ambos casos.



5.7 Interfaz final

6. Diseño e implementación

En este capítulo se describe la arquitectura utilizada, así como el análisis, diseño e implementación de la base de datos y del software que conforma la aplicación.

6.1 Arquitectura del modelo de capas

A continuación, se muestra una visión general de la arquitectura de la aplicación, exponiendo cada una de las partes que la integran, las cuales serán analizadas en profundidad en los siguientes apartados.

El software desarrollado se compone de tres capas o niveles, una capa de presentación, una de negocio y una tercera capa de persistencia.

La capa de presentación es aquella que visualiza el usuario cuando utiliza la aplicación, también conocida como interfaz gráfica. En ella se ofrecen los datos relevantes y se muestran las opciones disponibles, además de obtener del usuario la información necesaria para que el software pueda desempeñar la tarea para la que ha sido generado. En el capítulo cinco se han descrito en profundidad cada una de las interfaces gráficas de las que dispone el jugador de esta aplicación. Se han creado buscando simpleza y claridad, con el objetivo de que seas intuitivas para el usuario.

Dicha capa está dividida en dos partes. La primera está compuesta por componentes propios de Unity, donde existen unos objetos denominados *canvas* que recrean la interfaz inicial de la aplicación. También es el lugar donde se establece la cámara de la escena que hace referencia a Vuforia para el reconocimiento de imágenes. La segunda parte es la interfaz gráfica creada en C#, que contiene una función *OnGUI* que realiza actualizaciones periódicas para percibir si se ha modificado algún dato en las estructuras enlazadas al proceso.

Por otro lado, en esta capa se encuentra la clase que recibe los objetos reconocidos por Vuforia para su tratamiento, la cual es la encargada de comunicarse con los niveles adyacentes y realizar el flujo de ejecución que se explicará en las siguientes secciones.

La capa de presentación tiene la función de comunicarse con la siguiente capa mediante el controlador y los servicios de aplicaciones generados.

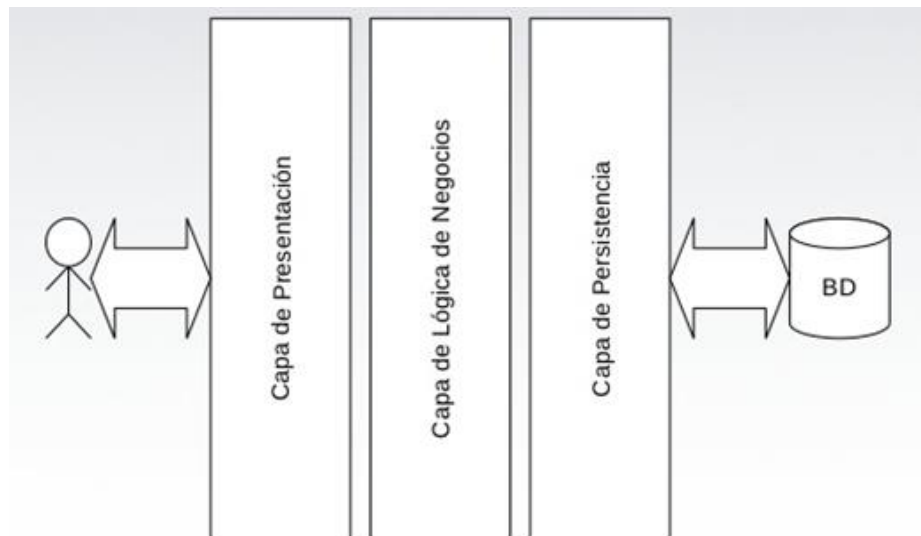
La capa de negocio contiene toda la lógica del proceso, controlando la entrada de peticiones hacia la capa de persistencia y administrando la salida de información a través de la capa de presentación. Se encarga de “analizar” el estado del programa en todo momento, comprobando si el usuario ha alcanzado alguno de los hitos que lo componen.

En este nivel se sitúan los objetos de transferencia de datos empleados para almacenar la información y utilizados para el tránsito entre capas. Se hace uso de un objeto para almacenar los datos relacionados con los elementos de la colección del museo, se emplea un segundo objeto que contiene la organización de los pasillos y secciones, además se dispone de un tercer objeto con el grafo de distancias entre las secciones del museo.

También existen dos clases vinculadas cada una de ellas a un componente Unity específico. Se encargan de la persistencia de todos los datos necesarios para el correcto funcionamiento de la aplicación durante el proceso de ejecución.

Asimismo, en la capa de negocio se hallan los servicios de aplicaciones, que contienen el algoritmo de comparación, el de posicionamiento y la función encargada de generar los componentes Unity necesarios para el conjunto de datos de Vuforia. Cabe destacar que estos servicios han sido generados mediante la utilización de una factoría y el uso del patrón *singleton* para garantizar su unicidad.

La tercera y última capa del software es la de persistencia. Se compone de los recursos necesarios para acceder a los datos almacenados en el servidor que contiene la base de datos, así como de preparar la información obtenida a través de la capa de presentación y negocio, encapsularla y enviarla como petición al servidor. Esta capa, a diferencia de las otras dos, está distribuida físicamente entre el hardware en el que se ejecuta la aplicación y el servidor que contiene los datos. En este nivel se encuentran los objetos de acceso a datos (DAO), encargados de realizar las peticiones al servidor si fuera necesario y transferir el resultado a la capa de negocio. Existen tres objetos de acceso a datos, uno para los objetos del museo, otro para el mapa y el último para los conjuntos de datos de Vuforia.



6.1 Arquitectura de capas

Toda la información referida a los objetos está almacenada en una base de datos no relacional MongoDB. Se encuentra alojada en un servidor Windows de la Facultad de Informática de la Universidad Complutense de Madrid. El acceso a dichos datos se realiza mediante peticiones desde el cliente de la aplicación.

El cliente en esta arquitectura es el dispositivo móvil que contiene instalado el software donde se establece la lógica de funcionamiento. Se hace uso de internet para enlazar el dispositivo móvil y el servidor, cargando al inicio de la aplicación todos los datos, generando las estructuras e inicializando el juego.



6.2 Esquema de arquitectura

Pese a la simplicidad que refleja la arquitectura nos ha proporcionado todo lo necesario para crear una aplicación que satisficiera todos los objetivos marcados sin ser esto un inconveniente.

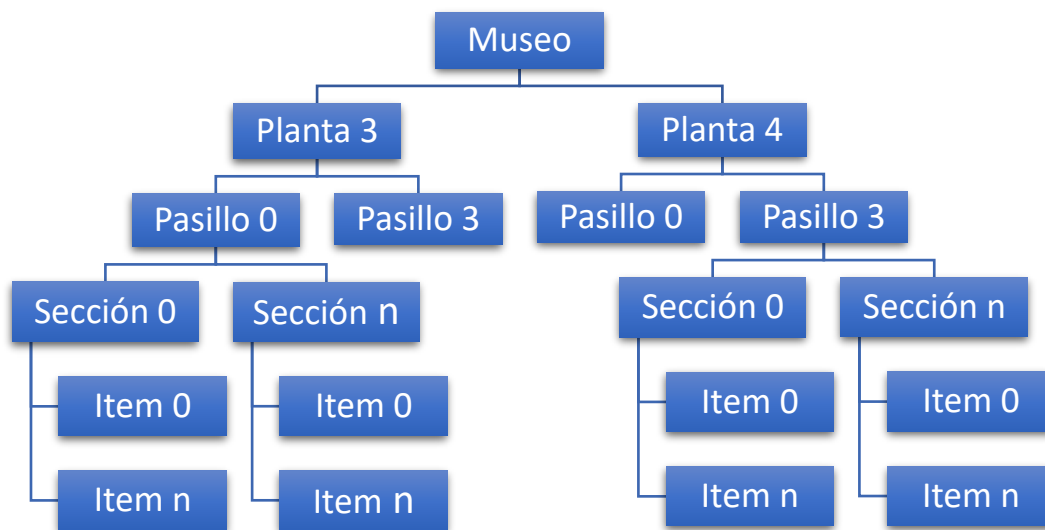
6.2 Base de datos de la aplicación

En esta sección se detallará el proceso de análisis, diseño e implementación de la base de datos empleada para realizar este proyecto. El sistema de datos generado que especificaremos a continuación sirve de pilar básico para el funcionamiento de los algoritmos detallados en el siguiente apartado.

6.2.1 Análisis y diseño de la base de datos

El proceso de análisis comenzó con la investigación de la disposición de los elementos del museo, donde se estableció la necesidad de generar un orden lógico para poder posicionar cada objeto unívocamente.

Se creó una ordenación basada en la división de plantas, pasillos, secciones y elementos. Se obtuvo como resultado de ello dos plantas, tercera y cuarta de la Facultad de Informática, cada una de ellas se dividió en cuatro pasillos y estos en secciones. Cada sección agrupaba un número de elementos del museo.



6.3 Jerarquía de ordenación del museo

Con esta ordenación se procede a nombrar cada objeto del museo, para lo que se utiliza la siguiente nomenclatura; “pl” denomina las plantas, “ps” los pasillos, “s” las secciones e “itemX” el elemento número “X” dentro de esa sección. Un ejemplo de esta nomenclatura es el siguiente: pl3_ps2_s4_item2 en el que

establecemos que dicho objeto se encuentra en la plata tres, pasillo dos, sección cuatro y segundo elemento empezando de derecha a izquierda y de abajo a arriba.

A continuación, se obtuvieron cuatro imágenes de cada uno de los elementos del museo, las cuales se nombraron con la nomenclatura establecida anteriormente. Para identificar cada una de las imágenes con el elemento del museo correspondiente se añadió una letra, A para la primera imagen, B para la segunda y así sucesivamente hasta completar todas, dando como resultado pl3_ps2_s4_item2A. En la siguiente imagen se pueden observar dos miniaturas de las fotografías en distintos ángulos obtenidas para uno de los objetos del museo.



6.4 Elemento del museo

El siguiente paso fue establecer el tipo de base de datos que se emplearía para el desarrollo de la aplicación. Se generó una discusión entre los beneficios e inconvenientes que proporcionaban las bases de datos relacionales frente a las no relacionales.

Las bases de datos relaciones tenían a favor que el equipo de proyecto conocía el funcionamiento del motor SQL. También se consideró plausible establecer un conjunto de atributos comunes para todos los elementos del museo, lo cual invitaba al uso de este tipo de base de datos. Sin embargo, durante una reunión con los tutores en la que se plantearon nuestras ideas a favor del uso de una base de datos relacional, se analizó más en profundidad y se determinó que nuestra premisa de establecer un esquema de atributos comunes no era correcta, puesto que las características de los objetos podrían variar en función de las aplicaciones finales del software. Esta idea unida a que las características de escalabilidad,

flexibilidad y rendimiento de una base de datos no relacional eran más adecuadas para el desarrollo de la aplicación motivaron que se tomase la decisión de elegir una base de datos no relacional, concretamente MongoDB.

Por otro lado, el proceso de investigación realizado al comienzo del análisis también esclareció la falta de documentación existente sobre los objetos del museo y la necesidad de obtener información de ellos. Debido al gran número de piezas que integran el museo se tomó la decisión de documentar una muestra de objetos de la planta tercera. Se determinó que la variedad de elementos de la muestra podría ser beneficiosa para el funcionamiento que se deseaba conseguir porque agrupaba diversos conjuntos, cuyos elementos tenían similitudes, lo que facilitaba que el algoritmo de comparación descrito en la siguiente sección arrojase resultados prácticos.

Con la información obtenida se crearon estructuras de datos de tipo *json*, una por cada elemento del museo. Se consideró emplear *jsons* porque facilitaban la composición de diferentes subconjuntos de atributos, debido a que cada grupo de elementos tenía unas características específicas. Por ejemplo, las memorias de almacenamiento tienen como atributos el peso o el tamaño entre otros, características que podemos encontrar en diferentes elementos del museo como los teclados o calculadoras, pero también tienen otras cualidades como la velocidad de escritura propia solo de este componente.

6.2.2 Atributos de la base de datos

Los atributos contenidos en los archivos *json* sufrieron varias modificaciones a lo largo del desarrollo de la aplicación. Se comenzó con una estructura de datos muy simple que contenía como primer campo un identificador *id* que seguía la nomenclatura mencionada anteriormente (*pl3_ps0_s11_item0*) para que de esta manera se relacionasen las imágenes con su estructura de datos correspondiente. Después se establecieron un conjunto de atributos relacionados con las características del objeto que a continuación se enumeran:

- Nombre del elemento.
- Modelo, que en caso de ser un objeto único recibe el nombre de prototipo.
- Año de fabricación.
- Lugar de producción.
- Breve descripción de la composición y funcionalidad del objeto.

- Tipo. Se estableció una categorización de los elementos de la muestra. Cada objeto puede tomar como máximo un indicador dentro del conjunto de valores preestablecidos, entre los que se encuentran computador, componente, terminal o consola.
- Subtipo. Con el fin de clasificar los elementos con mayor detalle se decidió encuadrarlos a todos dentro de unas subcategorías, por ejemplo, para los objetos de tipo componente las subcategorías decretadas son placa, teclado, ratón, memoria de almacenamiento o circuito integrado entre otros.
- Color. Los más comunes son negro, blanco, gris oscuro y gris crema.
- Tamaño. Se generó una clasificación con una escala desde uno hasta cuatro. El valor uno representa a los objetos muy pequeños, por ejemplo, los transistores; el número dos es para objetos pequeños como algunas consolas, el tres es para elementos medianos y el cuatro sirve para representar a los objetos muy grandes como, por ejemplo, los computadores analógicos más antiguos.
- Precio, representado en dólares.

Se puede observar en la siguiente imagen un ejemplo de este tipo de *json* con los atributos descritos.

```
{
  "_id": "p13_ps0_s11_item0",
  "object": {
    "name": "Ordenador Personal IBM XT",
    "model": "IBM PC XT Segunda generación",
    "age": "1983",
    "country": "Armonk, New York, EE.UU.",
    "description": "Segundo ordenador personal creado por IBM. Primero en incluir un disco duro. Modificó el estandar al retirar la interfaz para cassettes. Enfocado para negocios.",
    "type": "Computador",
    "subtype": "PC",
    "color": "Gris crema",
    "size": "3",
    "price": "4995"
  }
}
```

6.5 Estructura de datos inicial

Se consideró que todos estos atributos eran suficientes para que el usuario obtuviera información útil de los objetos que está visualizando en el museo a través de la aplicación. Pero al realizar las primeras pruebas se vio claramente que el número de campos que el algoritmo comparador (explicado en la siguiente sección en detalle) podía emplear para trabajar eran muy escasos, debido a que es necesario que los valores de los atributos de distintos objetos coincidan para

obtener resultados. Teniendo en cuenta que el nombre, el modelo y la descripción nunca toman valores iguales entre dos objetos, determinamos que era imprescindible aumentar el número de atributos para cada elemento.

Se generó una evolución del *json* mostrado anteriormente donde se añadió información acerca de la empresa a la que pertenece cada producto, específicamente su nombre, el nombre del fundador o fundadores, el año de creación y lugar donde se constituyó dicha empresa. Con estos campos el algoritmo tenía muchas más probabilidades de encontrar coincidencias entre objetos.

La siguiente fotografía ilustra una de estas estructuras de datos con los campos descritos.

```
{
  "_id": "pl3_ps0_s11_item0",
  "object": {
    "name": "Ordenador Personal IBM XT",
    "model": "IBM PC XT Segunda generación",
    "age": "1983",
    "country": "Armonk, New York, EE.UU.",
    "description": "Segundo ordenador personal creado por IBM. Primero en incluir un disco duro.
    | Modificó el estandar al retirar la interfaz para cassettes. Enfocado para negocios.",
    "type": "Computador",
    "subtype": "PC",
    "color": "Gris crema",
    "size": "3",
    "price": "4995"
  },
  "maker": {
    "number": "1",
    "name0": "International Business Machines - IBM",
    "founder0": "Charles Ranlett Flint",
    "founded0": "1911",
    "country0": "New York, New York, EE.UU."
  }
}
```

6.6 Estructura de datos evolucionada

Después de una reunión para establecer como se iniciaría el juego se consideró dar al usuario una pista inicial basada en una curiosidad del objeto que debe encontrar. Para satisfacer este requisito se tuvo que añadir a todas las estructuras de datos un apartado donde se almacenaban tres curiosidades de cada objeto.

Esto dio como resultado el siguiente *json*, en el cual se pueden visualizar todos los atributos que representan a un objeto. Como es lógico no todos los objetos tienen todos los atributos.

```
{
  "_id": "pl3_ps2_s11_item5",
  "object": {
    "name": "Ordenador Personal Commodore 64",
    "model": "Commodore 64",
    "age": "1982",
    "country": "EEUU",
    "description": "Ordenador personal de 8 bits. Es el modelo más vendido de computadora personal de la historia. Obtenía su nombre de los 64kB de memoria RAM de los que disponía. Debido a que estaba enfocado al espectro económico del mercado y a la modificación de su precio durante los primeros meses, este PC se consolidó como una de las opciones más populares de manera muy rápida.",
    "type": "Computador",
    "subtype": "PC",
    "color": "Gris crema",
    "size": "2",
    "price": "1477"
  },
  "curiosities": {
    "num0": "A pesar de ser un ordenador multipropósito más de la mitad del software desarrollado durante los primeros años era de ocio y videojuegos.",
    "num1": "Tan solo la familia de consolas Atari podía rivalizar con el C64 en sus características gráficas o de sonido, dejando en evidencia al Apple II o al VCS.",
    "num2": "No empleaba un sistema operativo como tal sino una versión de BASIC 2.0 limitada a la que se le habían retirado algunos comandos de control."
  },
  "maker": {
    "number": "1",
    "name0": "Commodore International",
    "founder0": "Jack Tramiel",
    "founded0": "1954",
    "country0": "EE.UU."
  }
}
```

6.7 Estructura de datos final

También se creó una única estructura de datos adicional para almacenar el posicionamiento de los elementos en el museo. Para ello se generó un *json* donde se especifica para cada *id* la posición que ocupa dentro del museo.

6.2.3 Implementación de la base de datos

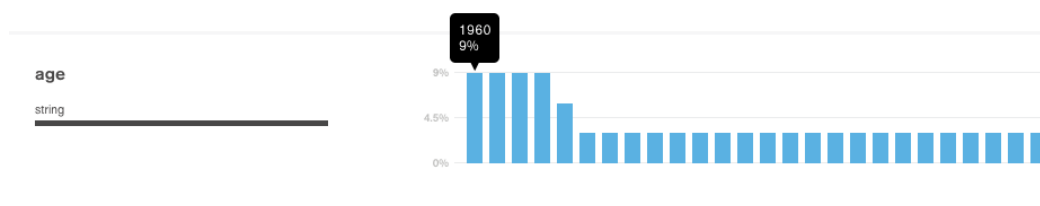
Se utilizó MongoDB como base de datos empleada para el almacenamiento de la información debido a la imposibilidad de tener un esquema de datos único.

Al finalizar la creación de las estructuras de datos, se procedió a generar con la herramienta *Mongo DB Compass* una única colección de registros en la que se aglutinó la totalidad del museo facilitando el acceso a la información mediante consultas MongoDB, que permiten desde obtener la colección completa de datos hasta recuperar el número de objetos que contienen un atributo y un valor específicos.

Además de permitir la gestión de la base de datos, *Mongo DB Compass* nos provee de diferentes estadísticas en función de los datos de la colección, las cuales nos han ayudado a determinar que el algoritmo de comparación detallado en el siguiente capítulo, tiene el comportamiento deseado, debido a que si la dispersión de los valores es menor el algoritmo tiene mayores probabilidades de devolver coincidencias.

Como se ha detallado en la sección de análisis y diseño de la base de datos, se tomó de todos los objetos del museo una muestra de la planta tercera para ser documentada. Al obtener las gráficas de esta primera muestra se observó que la dispersión de los valores era muy alta, es decir, casi todos los elementos de la muestra tenían valores distintos para atributos determinados, lo que hacía difícil encontrar coincidencias a la hora de comparar.

En la siguiente gráfica podemos ver un ejemplo de esto. Se puede observar que hay cinco años donde coinciden en la fabricación dos o más objetos, pero el resto han sido producidos en años diferentes, por lo tanto, no se obtienen prácticamente coincidencias en este atributo a la hora de compararlo.

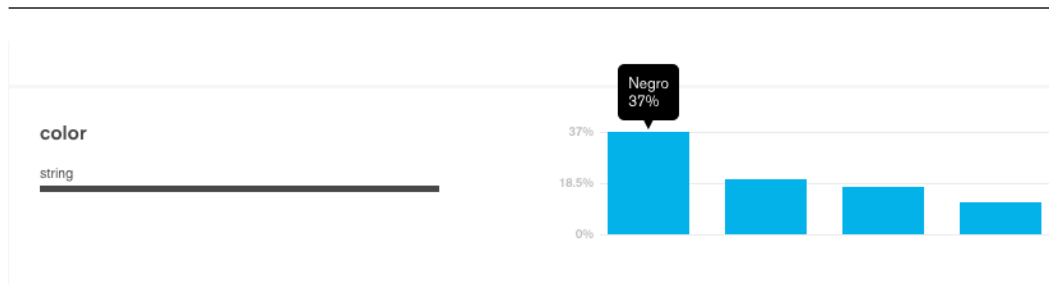


6.8 Estadística Mongo DB Compass I

Este hecho determinó la necesidad de aumentar la muestra de objetos seleccionados para ser documentados. Se procedió a elegir elementos del museo que beneficiasen una disminución de la dispersión de los valores.

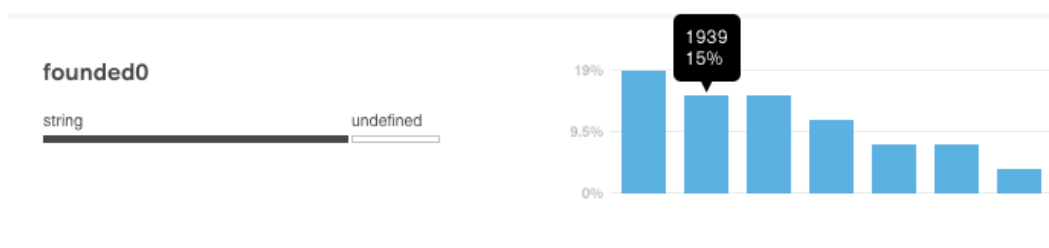
Al obtener nuevamente las gráficas de datos de esta segunda muestra observamos que se habían corregido en gran medida estos problemas, como podemos observar en las siguientes imágenes.

En esta primera se puede visualizar que el color de los objetos se encuentra prácticamente repartido entre solo cuatro valores, siendo el color negro el que tienen un mayor porcentaje de objetos.



6.9 Estadística Mongo DB Compass II

El año de fundación de las empresas, aunque más repartido que el color, también muestra que hay varios valores en los que coinciden un buen porcentaje de los objetos. Se puede visualizar en la gráfica que un quince por cien de los objetos fueron fabricados por empresas fundadas en el año 1939. También se puede apreciar en la barra horizontal (undefined) que no todos los objetos tienen este atributo.



6.10 Estadística Mongo DB Compass III

En esta última gráfica en la cual aparece el porcentaje de objetos creados por una empresa también podemos observar que los valores no tienen una alta dispersión.



6.11 Estadística Mongo DB Compass IV

Una vez solucionados los problemas de dispersión se tomó la decisión de alojar la base de datos en un servidor de la Facultad de Informática para mejorar la

eficiencia de acceso, la seguridad y la mantenibilidad de los datos. Esta decisión se llevó a cabo teniendo presente también el hecho de que se trata de un juego y no se deseaba que los datos relevantes a la mecánica del mismo quedasen a disposición del usuario.

Cabe destacar que se ha implementado un servicio de *Windows* que permite iniciar la base de datos de forma automática. Este proceso está detallado en el manual de instalación de esta memoria, situado en el apéndice final.

6.3 Diseño de la aplicación

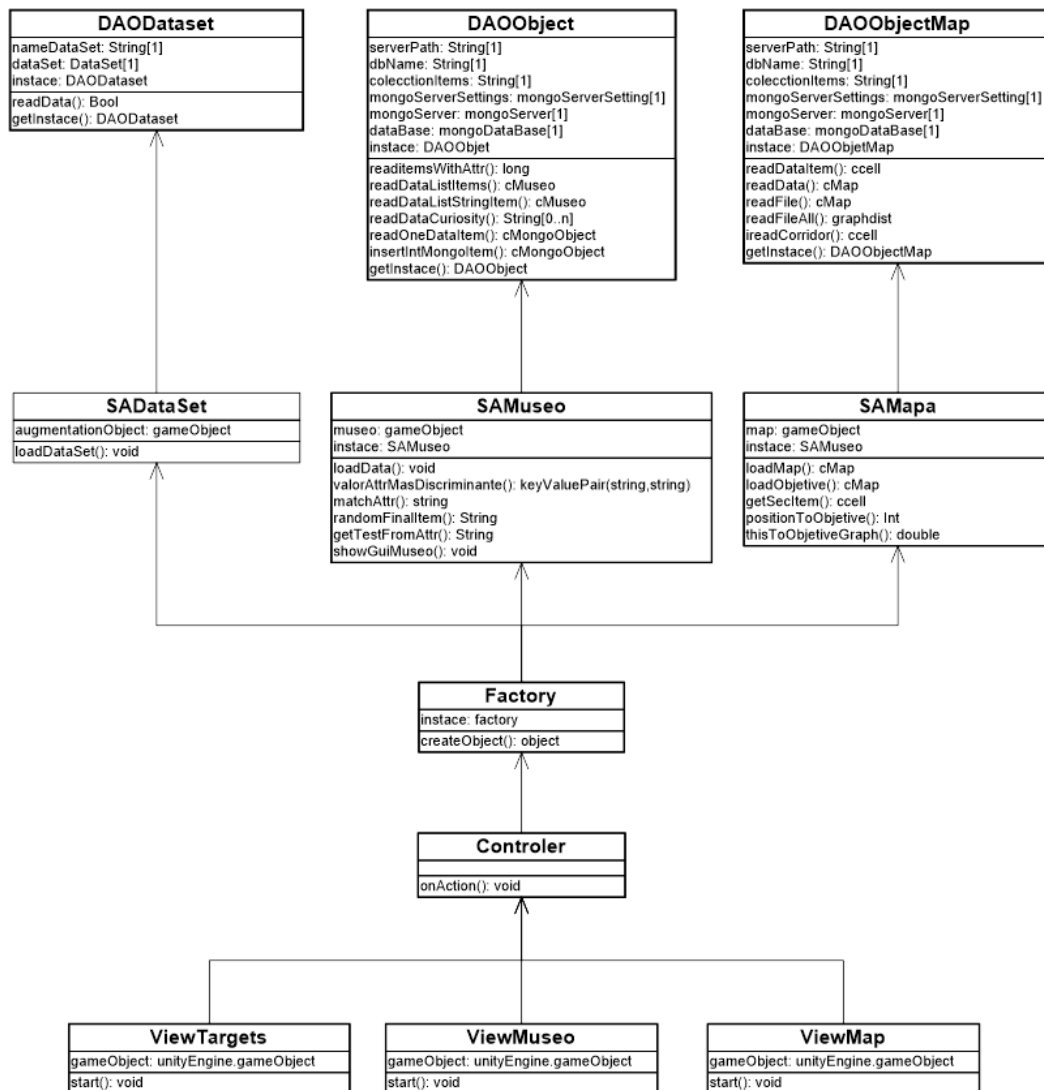
En esta sección se aborda el diseño del software de la aplicación móvil. Como se ha mencionado anteriormente, se ha hecho uso de la plataforma Unity en colaboración Vuforia para posibilitar la RA.

En la figura 6.12 presentada a continuación, se puede visualizar el diagrama del diseño de la aplicación, donde se puede percibir la estructura que se ha seguido.

En la capa de integración están incluidos los DAOs, los cuales posibilitan el acceso a los datos bajo petición de los servicios de aplicación. El DAO de Objetos y el del Mapa se conectan al servidor MongoDB para la obtención de datos, mientras que el DAO *dataSet* obtiene los archivos de Vuforia necesarios incluidos como recursos del software en el dispositivo móvil.

Los tres servicios de aplicaciones contienen los métodos necesarios para el funcionamiento de la aplicación. Los servicios, aparte de hacer uso de los DAOs, también utilizan los objetos de transferencia de datos y componentes Unity para la presentación y resolución de las funciones que se generan en la aplicación.

La factoría hace uso de los servicios de aplicación para su creación. Dependiendo de los parámetros que reciba la factoría creará un objeto u otro. Dichas variables son enviadas mediante el *viewMap*, *viewMuseo* y *viewTarget* que aparte de ser clases que hacen uso de la factoría son componentes de Unity accionados al comienzo de la ejecución de la aplicación.



6.12 Diagrama de diseño

6.3.1 Flujo de ejecución

Para cumplir con el funcionamiento acordado, el cual se ha ido moldeando y acotando en las diversas reuniones de seguimiento del proyecto, se especifica el flujo de ejecución con el fin de obtener una visión clara del funcionamiento de la aplicación.

Una vez que se inicia la aplicación, se cargan mediante una petición al servidor todos los objetos del museo y el mapa del mismo. Después de tener la información almacenada se obtiene el objeto final de forma aleatoria, con este y con los demás objetos se realiza un grafo de conexión y distancias entre todos ellos basándonos en el mapa.

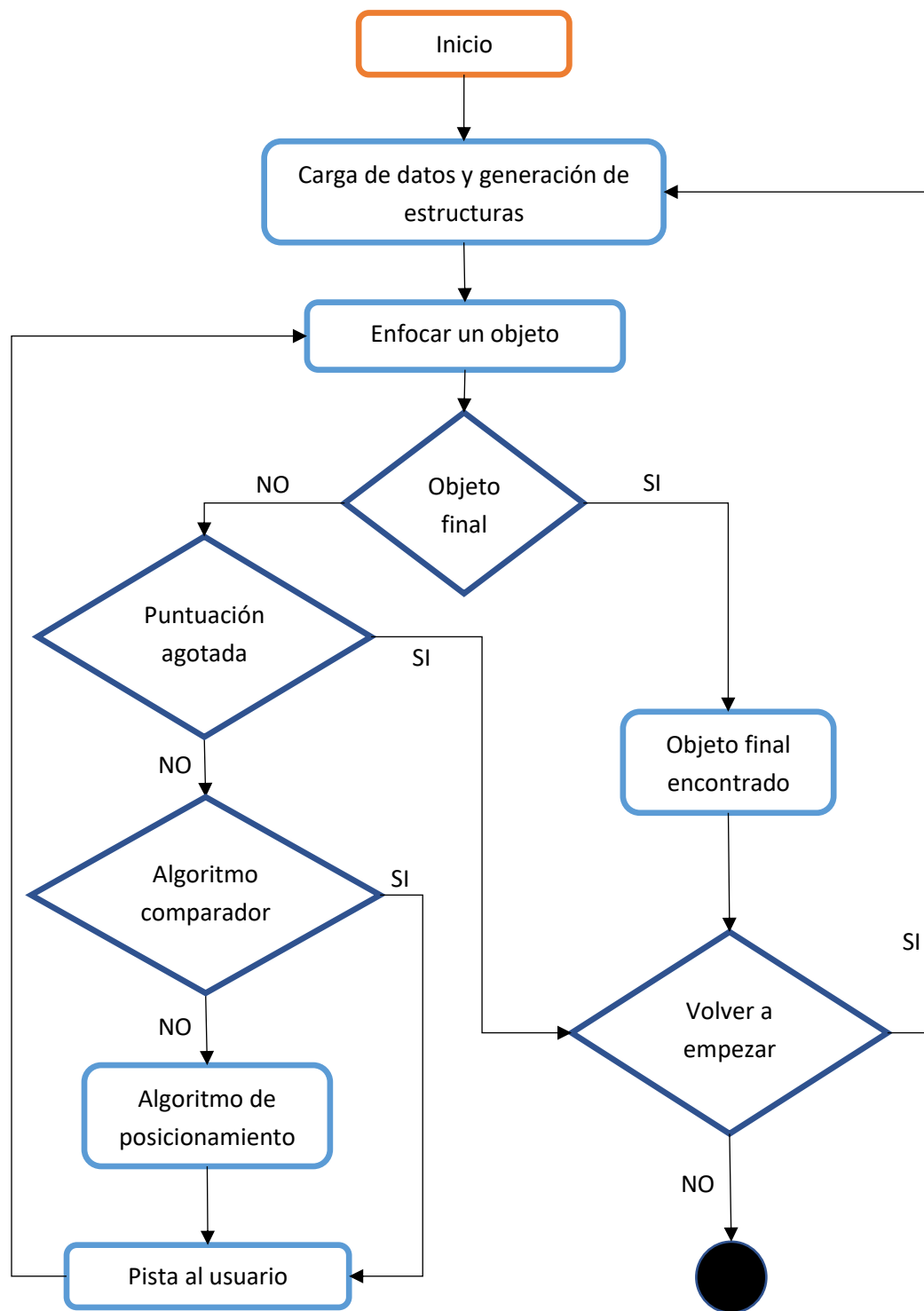
La aplicación reacciona a las acciones del usuario y cuando éste enfoca un objeto, se pone en marcha toda la lógica del software. Una vez reconocida la imagen, se observa si es el objeto final. En caso negativo se comprueba si la puntuación del usuario es menor o igual a cero. Si la puntuación es positiva se ejecuta el algoritmo comparador, el cual nos puede proporcionar una pista de relación entre el objeto enfocado y el objeto final o en caso de que no exista ninguna relación llamar al algoritmo de posicionamiento. En ambos casos el usuario obtendrá una pista que le llevará a enfocar otro objeto para seguir recibiendo información o llegar al objeto final.

Si al enfocar un objeto el software detecta que es el elemento buscado, el usuario habrá ganado, se mostrará su puntuación y se le dará la posibilidad de volver a jugar o salir del juego. En contraposición, existe la posibilidad de que al enfocar un objeto este no sea el buscado y la puntuación pase a ser negativa. En dicho caso el juego habrá terminado y el usuario habrá perdido, por lo que se le mostrará la opción de volver a jugar o salir.

En un principio se dependía del posicionamiento para iniciar la aplicación, es decir, era necesario estar en un punto concreto del museo para que funcionara correctamente, esto limitaba la libertad del jugador, por ello se tomó la decisión de modificarlo y permitir que el usuario iniciase el juego en la posición que deseara.

Otro cambio importante fue establecer el objeto final de forma aleatoria. En inicio, el usuario tenía la posibilidad de elegir una categoría en la que estaban encuadrados los objetos y a partir de esa categoría se obtenía un objeto final. Este funcionamiento se modificó para generar un uso más sencillo y fluido para el usuario.

El flujo de ejecución está compuesto por procesos más simples que explicaremos a continuación.



6.13 Flujo de ejecución

Carga de datos y generación de estructuras

Este proceso se encarga de obtener toda la información de la base de datos mediante una petición en el DAO contenido en la capa de integración. Se obtienen los datos de los elementos del museo y del mapa, almacenándose todo ello en objetos de transferencia de datos para su uso en la ejecución de la aplicación. Concretamente se emplean cadenas de datos (clave-valor) en las cuales se pueden incluir los atributos obtenidos de la petición sin una estructura concreta. Para la recogida de los datos del mapa se usan variables que puedan contener un número ilimitado de pasillos y celdas.

En este punto se generan los componentes Unity del museo con la ayuda del conjunto de datos de Vuforia que contiene el mapa de puntos de cada objeto. Dicho componente reacciona al reconocer el objeto mediante la cámara del dispositivo móvil.

La información obtenida previamente de cada elemento de la base de datos se relaciona con los componentes Unity para que tengan asociados el *id*, nombre, descripción y mapa de puntos de cada elemento al que referencian.

Por otro lado, el mapa de objetos del museo permite generar un grafo con las secciones y distancias entre todos los componentes del mismo. Este algoritmo de generación está concebido concretamente para la tipología de las plantas de la colección García Santesmases en la Facultad de Informática de la Universidad Complutense de Madrid.

Objeto final y puntuación agotada

En este proceso la clase encargada de tratar los objetos reconocidos por Vuforia, ubicada en la capa de presentación, realiza dos comprobaciones.

La primera comprobación se encarga de verificar si el elemento enfocado es el objeto final buscado. Si el resultado es positivo la ejecución continuará en el proceso “volver a empezar”. En caso contrario, si es negativo, se realiza la segunda comprobación.

La siguiente verificación consiste en determinar si el usuario ha agotado sus puntos. En caso de que así sea, el flujo de ejecución continuará en el proceso

“volver a empezar”. Si el usuario aún conserva alguno de sus puntos se procederá a la ejecución de los siguientes algoritmos.

Algoritmo comparador

El algoritmo comparador se encuentra en la capa de negocio de la aplicación, concretamente en el servicio de aplicaciones del museo. Se comunica con la capa de persistencia mediante el DAO para la obtención de los objetos de transferencia de datos necesarios para su funcionamiento.

Este proceso se encarga de obtener una pista mediante el cotejo de los atributos de los objetos. Como ya se ha explicado con anterioridad, cada objeto tiene una serie de atributos propios en la base de datos que pueden o no coincidir con el resto de objetos. Este algoritmo requiere dos parámetros de entrada, el objeto enfocado y el objeto final.

Al comienzo el algoritmo obtiene los atributos y valores de cada parámetro de entrada, los compara y extrae los comunes con el mismo valor. Una vez que tiene la lista de parejas (atributo, valor) coincidentes, se procede a buscar con esa lista en el resto de objetos del museo. De esta forma se obtendrá una lista ordenada, en forma decreciente, de los atributos más comunes de toda la colección.

La lista generada nos permite extraer el primer elemento y verificar que la pista no ha sido mostrada anteriormente al usuario cuando visualizó otro objeto, y en caso de haberla mostrado se elige la siguiente pista dentro de la lista ordenada. Si se llegase a la situación de haber mostrado todas las pistas contenidas en la lista el algoritmo no devolverá nada, al igual que si no se hubiera encontrado una lista de pistas coincidentes.

La complejidad del algoritmo es alta, el tiempo medio de ejecución del mismo está en el orden del número de atributos de los objetos del museo (A) y el número de objetos del museo (N). Por tanto, el algoritmo tiene un tiempo medio de ejecución de $O(N \times A)$.

La lista de atributos coincidentes se ordena de forma decreciente con el objetivo de mostrar en primer lugar el atributo más frecuente entre los objetos del museo. De esta manera se aumenta la dificultad para el usuario que recibe las pistas a la hora de encontrar el objeto final.

Como ejemplo de funcionamiento se presenta el siguiente caso, en el apartado anterior se mostró una estadística de la base de datos con el color, figura 6.9. Como se puede observar en la gráfica todos los objetos del museo poseen este atributo, por tanto, se puede establecer que el algoritmo comparador siempre se ejecutará para comprobar si los objetos coinciden en su valor.

Se establece como premisa para este ejemplo que el valor del atributo color del objeto enfocado y del objeto final es el mismo, concretamente el negro, sino fuera así no se harían las siguientes comprobaciones.

Se procede a buscar el número de objetos de la base de datos que tienen esta misma pareja (color, negro), como resultado se obtiene que existen un 37% de objetos del museo con esta característica y se clasificará en la lista de atributos coincidentes con respecto al resultado en orden descendente. Si al finalizar el algoritmo el atributo color se encuentra en la primera posición de la lista de coincidencias y no ha sido mostrado con anterioridad, se mostrará como pista dicho color mediante la capa de presentación de la aplicación.

Algoritmo de posicionamiento

Al igual que el algoritmo anterior, este se encuentra en la capa de negocio de la aplicación, concretamente en el servicio de aplicaciones del mapa. De igual manera se comunica con la capa de persistencia mediante el DAO para la obtención de los objetos de transferencia de datos necesarios para su funcionamiento.

Este algoritmo se encarga de obtener una pista de posición para el usuario, para ello necesita conocer la sección en la que se encuentra el objeto enfocado y el objeto final. También se utiliza el mapa del museo y el grafo previamente generado.

Con la ayuda del mapa se obtiene de manera sencilla dónde se encuentra la sección enfocada con respecto a la sección final. Como se tiene una estructura con los pasillos y secciones de todo el museo, se recorre hasta encontrar el objetivo. Los objetos poseen en su nombre la sección y pasillo a la que pertenecen y el algoritmo se apoya en el objeto enfocado para la búsqueda en la estructura. El usuario recibe una dirección a la que dirigirse, la cual se encuentra dividida en cuatro indicaciones:

- Delante, el usuario puede ir a un pasillo que se encuentre físicamente frente a él.
- Detrás, el usuario puede ir a un pasillo que se encuentre a su espalda.
- Derecha o izquierda, el usuario puede buscar un objeto que se encuentre en dichas direcciones.

El jugador también recibe los metros restantes al objetivo final. Usando el algoritmo de *Dijkstra* sobre el grafo se obtiene la suma de las distancias entre los vértices (secciones) del camino, desde el elemento enfocado hasta el objeto final.

Los resultados obtenidos por el algoritmo se devuelven a la capa de presentación para mostrarlos al jugador.

Dar pista al usuario

Esta funcionalidad se encarga de mostrar al usuario la pista obtenida en los procesos anteriores. Como se ha mencionado, los resultados devueltos por los algoritmos se envían desde la capa de negocio a la capa de presentación, para que puedan ser visualizados por el jugador.

Para las pistas obtenidas mediante el algoritmo comparador existe un texto preestablecido por cada atributo registrado que ofrece una mayor coherencia al visualizarlo en pantalla, para el resto de atributos se muestra un texto genérico. Además, se mantiene un registro de las pistas mostradas para su utilización en dicho algoritmo que permite revelarlas al usuario en la interfaz de la aplicación junto con el número de visitas a cada objeto.

Las pistas de posicionamiento informan al usuario de la distancia en metros y la dirección en la que se encuentra el objeto buscado.

Todas estas pistas se muestran unos segundos en pantalla y posteriormente la aplicación queda a la espera de procesar otro objeto.

Volver a empezar

Este proceso ubicado en la capa de negocio de la aplicación se encarga de eliminar el contenido de los componentes de Unity y de todos los objetos de transferencia de datos. El siguiente paso es arrancar el proceso que se ejecuta al iniciar la aplicación.

7. Conclusiones y trabajo futuro

7.1 Conclusiones

Llegados a este punto, último capítulo de la memoria, es hora de extrapolar los resultados obtenidos del trabajo realizado. Ha sido un duro camino en el que hemos ido adaptándonos a las necesidades y obligaciones que se presentaban para alcanzar los objetivos iniciales.

Cabe mencionar que durante este proceso se ha ido aprendiendo un nuevo lenguaje de programación (C#), junto con la utilización de una nueva plataforma de desarrollo llamada Unity y el reconocimiento de imágenes de Vuforia. Además, se ha trabajado con una base de datos desconocida por el equipo hasta la implementación en este proyecto (Mongo DB).

Con tiempo y esfuerzo se han ido adquiriendo los conocimientos necesarios en todas estas tecnologías para concluir este proyecto de manera satisfactoria.

Se puede afirmar que se han cumplido la mayoría de objetivos que se plantearon al inicio del camino:

- Se consiguió documentar un número suficiente de objetos del museo para nutrir de manera efectiva la base de datos y poder trabajar con ellos.
- Se ha logrado realizar un algoritmo para comparar los diferentes atributos de los objetos, el cual ha sido explicado en detalle en anteriores capítulos. Este algoritmo nos ha posibilitado interactuar con los elementos del museo.
- Se han superado las diversas dificultades que se presentaron con el posicionamiento, creando una alternativa eficiente al posicionamiento WiFi del que se partió, usando para ello un grafo y un mapa con ayuda del reconocimiento de imágenes.
- Se ha creado una interfaz sencilla, funcional y atractiva para el usuario, para ello ha sido necesario que el equipo aumentase sus conocimientos en Unity y se dedicase plenamente a las sucesivas versiones de la interfaz.

- Las tecnologías utilizadas se eligieron expresamente para permitir el cumplimiento de otro de los objetivos planteados en el inicio del proyecto, crear un software que posibilitara una arquitectura multiplataforma.

Consideramos que se ha cumplido el deseo de entretener a los usuarios interesado por la cultura gracias la combinación entre la educación y el juego.

Valoramos positivamente el trabajo realizado y estamos satisfechos con el estado final del proyecto debido al cumplimiento de los objetivos marcados al inicio del mismo. Consideramos que el software realizado tiene el suficiente potencial para evolucionar con nuevas funcionalidades en sucesivas versiones.

7.2 Trabajo futuro

A continuación, se detallarán las mejoras viables en base a las características del software desarrollado. Estas quedarían pendientes para las próximas versiones de la aplicación.

- Completar el registro de todos los objetos del museo García Santesmases y añadirlos a la base de datos con los actuales. Mejoraría la experiencia de juego al ofrecer un mayor número de elementos que visualizar.
- Mejorar las fotos de los objetos del museo para aumentar la potencia del reconocimiento de imágenes.
- Implementar otros tipos de pistas en base a diferentes comparaciones entre atributos de los objetos y aumentar la precisión de las indicaciones de dirección, todo ello proporcionará una mayor variedad de las ayudas al usuario que enriquecerá el juego.
- Añadir nuevas modalidades aumentando o disminuyendo la dificultad de las pistas a petición del usuario para ajustar la experiencia a sus conocimientos.
- Adaptar el proyecto a las características que proporcionen las nuevas versiones de Unity y Vuforia, para aprovechar las mejoras de software.

- Realizar un módulo de administración que facilite añadir y localizar en base de datos los objetos del museo sin necesidad de realizarlo manualmente. Esta mejora proporcionará mayor comodidad y rapidez en las tareas de mantenimiento.

8. Conclusions and future work

8.1 Conclusions

At this point, the last chapter of this document, it is time to extrapolate results obtained from the work done. It has been a difficult road in which we have been adapting our resources to the demands and obligations that were presented to reach the initial objectives.

It needs mentioning that during this process we have learned a new programming language (C#) along with the use of a new development platform (Unity) and Vuforia's recognition system. In addition, we have worked with an unknown database until the implementation of this project (MongoDB). With time and effort we have acquired the necessary knowledge in all these technologies to complete this project in a satisfactory way.

With time and effort we've been acquiring the knowledge required in all the technologies mention before to conclude this project in a satisfactory way.

It can be assured that most of the goals that were raised at the beginning of this road have been achieved.

It was possible to document a sufficient number of objects of the museum to effectively nurture the database and to be able to work with them.

We've been able to perform an algorithm to compare different attributes of the objects, which has been explained in detail in previous chapters. This algorithm has enabled us to interact with the elements of the museum.

The difficulties that were presented with the positioning system were overcome by creating an alternative to the Wifi positioning from which we started using a graph and a map with the support of an image recognition technology.

We have been able to create a simple interface, functional and attractive for the user in which was necessary for the team to master/ assimilate/ develop the required knowledge in the Unity system and fully commit to the interface following versions

The technologies used during this project were expressly chosen to allow the attainment of another of the objectives set during the start of the project: to create a software that would enable a multiplatform architecture.

We consider that we've achieved the wish to entertain the users interested in popular culture thanks to the combination between learning and playing.

We value favourably the work done and we are satisfied with the final stage of the project due to the fulfillment of the goals set during the opening phase. We consider the initial software to have enough potential to evolve with further functionalities in coming versions.

8.2 Future work

Next, the feasible improvements will be detailed based on the characteristics of the developed software. These would remain pending for the next versions of the application.

- To finish the registration of all objects of the García Santesmases museum and add them to the database with the currently existing ones. It would improve the gaming experience by offering more items to view.
- To improve photos of museum objects in order to increase the work of image recognition.
- To implement other types of clues based on different comparisons between attributes of the objects and increase the accuracy of the directions given. These things will provide a greater variety of user helps that will enrich the game.
- To add new game options, increasing or decreasing the difficulty of the clues at the user's request to adjust the experience to their knowledge.
- To update the project to the new versions of Unity and Vuforia, to take advantage of software improvements.

- To develop a management module that makes easier to add and locate objects in the museum, without having to do it manually. This improvement will provide greater comfort and speed in maintenance tasks.

9. Aportaciones personales

A continuación, se detallarán las aportaciones realizadas durante el transcurso del proyecto por cada miembro del equipo, especificando individualmente las tareas acometidas.

9.1 Juan José Arconada Alonso

El inicio de mi trabajo en este proyecto comenzó en junio del 2016 tras una reunión con el profesor D. Gonzalo Méndez. En ella se nos explicó de manera sucinta el objetivo inicial del proyecto y los desafíos técnicos que presentaba. Al terminar la misma, comenzó un proceso de investigación que incluyó el aprendizaje del funcionamiento de Android como plataforma para aplicaciones software.

Además de entender la mecánica de Android hizo falta aprender el funcionamiento de Unity, que apuntaba a ser el motor sobre el que desarrollaríamos la aplicación, por lo que durante los siguientes meses fue necesario visualizar tutoriales al respecto.

Llegado el inicio de curso se emprendieron sucesivas reuniones donde se establecieron los hitos del proyecto. Del mismo modo, se nos hizo entrega de materiales que podrían ser útiles, como código de posicionamiento, referencias de material bibliográfico de proyectos similares y el dispositivo móvil sobre el que trabajaríamos. Mi papel en esta parte se centró en analizar la documentación de esos proyectos y comprobar mediante pruebas de trabajo las capacidades del aparato del que disponíamos.

Apoyándome en el criterio de los profesores, llevé a cabo un trabajo de investigación sobre las posibles opciones de RA existentes en el mercado. Esto me sirvió más adelante para redactar parte del capítulo “Trabajo relacionado” de la memoria.

Una vez se discutieron en reuniones posteriores los problemas que nos encontraríamos y que motivaron la modificación de algunos objetivos se empezaron a establecer las tareas necesarias para llevar a cabo las primeras

iteraciones del proyecto. Mi esfuerzo en este punto se encaminó hacia la creación de la documentación necesaria de los objetos del museo, buscando referencias y datos interesantes que introducir en el programa.

Para proceder con esta tarea hizo falta anotar la gran mayoría de objetos que se encontraban en el museo, descartando los irrelevantes o que se escapaban del alcance de la aplicación por la dificultad que entrañaba hacer una criba eficiente de los mismos.

Una vez llevada a cabo esta tarea con la ayuda de mi compañero Roberto Moreno, se comenzó el trabajo de búsqueda de datos de los mismos; accediendo a archivos de la Facultad de Informática, a la web del Museo y, especialmente, empleando internet. Con los resultados obtenidos se generó un listado que fue evolucionando conforme íbamos siendo conscientes de las necesidades de la base de datos y del algoritmo de comparación.

Además de investigar sobre los elementos, se empleó este trabajo para plantear la estructura y la organización de la base de datos que contendría todo ello, decidiéndose finalmente tras un par de reuniones con los tutores utilizar una base de datos no relacional, MongoDB, lo que me obligó a aprender sobre la misma debido a la falta de experiencia previa.

Establecida la base de datos y con los objetos del museo inicializados parcialmente, procedí a realizar un algoritmo que otorgara al usuario una pista comparando los atributos de los objetos del museo. Este algoritmo fue definido y depurado a lo largo de diferentes reuniones hasta alcanzar el estado actual.

Por otra parte, una vez tuvimos creado el grueso del software (algoritmo búsqueda y comparación) se empezó a trabajar más en la interfaz gráfica del programa, muy rudimentaria hasta ese momento, para ello se emplearían bibliotecas nativas de Unity.

Junto con mis compañeros, y tras recibir un poco de orientación por parte de los tutores del proyecto, llevamos a cabo una investigación de lo que sería más conveniente para los usuarios. Mi parte aquí, análoga a la de ellos, consistió en plantear soluciones a los problemas de usabilidad buscando una mayor simplicidad para el usuario final.

Mi trabajo en la parte técnica de la aplicación termina llevando a cabo varias pruebas de campo recreando las condiciones naturales de ejecución de la

aplicación. Esto permitió identificar problemas con el algoritmo de comparación que se había llevado a cabo, ya que al ser un set de datos limitado las soluciones que mostraban eran demasiado similares.

En la parte correspondiente a la memoria, mi trabajo se ha ceñido a los capítulos iniciales (introducción, motivación, objetivos, estado del arte) así como a las traducciones completas de la lengua extranjera obligatoria. Este trabajo me ha obligado a investigar en profundidad sobre la RA, así como su historia y la situación actual de la misma. Para ello, y apoyándome en la idea de que no obtendría gran cantidad de documentación en formato físico, empleé Internet para acceder a la misma, leyendo *papers* al respecto y reportajes de revistas técnicas que me dieron una amplia visión de ella. Esto propició que se ampliase mi imaginación hacia posibles escenarios futuros para nuestra aplicación.

9.2 Roberto Moreno Rojas

En un inicio mantuvimos diversas reuniones con los directores del trabajo en las que empezamos a establecer los pilares de la aplicación. El posicionamiento WiFi, la realidad aumentada y la generación dinámica de pistas fueron los puntos más importantes en los que basarnos y recopilar información para dar comienzo al trabajo.

Mi primera tarea fue trabajar en colaboración con mi compañero Víctor Yagüe en el posicionamiento WiFi. Ambos realizamos diversas pruebas para establecer si era posible integrar todas las tecnologías necesarias para el funcionamiento de la aplicación. El resultado de dichas pruebas arrojó la imposibilidad de utilizar el posicionamiento WiFi y Unity con Vuforia, debido a que el *Manifest* de ambas tecnologías entraba en conflicto.

A partir de ahí, planteamos diferentes posibilidades para posicionar al usuario dentro del museo, determinando que la solución más beneficiosa era hacerlo mediante el reconocimiento de imágenes.

Mi siguiente cometido en este punto del desarrollo fue buscar documentación del Museo García Santesmases. Dicha búsqueda tuvo resultados infructuosos debido a la escasa información encontrada sobre el mismo.

El museo cuenta con una página web en la que aparece una breve presentación y una pobre descripción de ciento quince objetos en la que se visualiza una pequeña fotografía de cada uno de ellos y algunos datos.

Debido a que rápidamente aprecié la falta de información necesaria, decidí proceder a documentar, en colaboración con mi compañero Juan Arconada, el museo para que fuera posible garantizar el correcto funcionamiento de la aplicación.

Para subsanar estas carencias, en primer lugar, ordené el museo por plantas, pasillos, secciones y objetos, como se explica detalladamente en el capítulo sexto de la memoria, otorgando un nombre único a cada objeto. Para ello utilicé esta nomenclatura `pl3_ps1_s1_item6`.

El siguiente paso fue averiguar cómo funcionaba el reconocimiento de imágenes de Vuforia. Para ello realicé cuatro fotografías en diferentes perspectivas de cada uno de los objetos existentes en el museo, con el fin de optimizar el reconocimiento de imágenes basándome en el funcionamiento aprendido.

Retoqué digitalmente cada fotografía, en total unas seiscientas, para obtener imágenes nítidas de cada elemento del museo. Esta tarea fue compleja debido a que las urnas de cristal en las que se encuentran expuestos los objetos provocan brillos con el reflejo del sol o el flash de la cámara.

Pasé a ordenar las imágenes, identificarlas unívocamente y relacionarlas con el objeto al que pertenecen, usando la nomenclatura establecida anteriormente seguida de una letra, la A para la primera imagen, la B para la segunda y así sucesivamente hasta completar las cuatro imágenes por cada objeto (Ejemplo: `pl3_ps1_s1_item6A`).

Una vez obtenidas y categorizadas todas las imágenes realicé pruebas generando diferentes *dataset* para conocer el funcionamiento de Vuforia. Conjuntamente efectué numerosas comprobaciones para determinar satisfactoriamente el correcto funcionamiento del reconocimiento de imágenes.

Una vez realizado todo el procedimiento anterior, el siguiente avance en el trabajo fue crear un mapa donde posicionar los objetos, para lo que generé un *json* en el que almacenar la posición de cada elemento del museo utilizando la ordenación anteriormente mencionada para las imágenes.

Colaboré con Víctor Díaz para crear un algoritmo de posicionamiento, generando para ello un grafo con las posiciones y distancias entre objetos a partir del *json*.

En diversas reuniones con los tutores decidimos utilizar una base de datos no relacional, por lo que MongoDB fue la opción elegida, la cual alojé en un servidor de la Facultad de Informática.

Debido al gran número de elementos del museo decidimos crear estructuras de datos sólo para los objetos de la tercera planta, que fueron inicializadas con toda la información recogida anteriormente en la documentación del museo.

Por todo ello, la estructura de datos actual es resultado de sucesivas evoluciones en las que se determinó por parte del equipo de trabajo que cuanto mayor fuera el número de datos de cada objeto del museo mejor funcionaría el algoritmo comparador.

Además, realizando pruebas deduje que se necesitaba una muestra mayor de elementos para que los resultados arrojados por el algoritmo comparador fueran los deseados, por lo que decidí incrementar el número de objetos en la base de datos.

Por otro lado, efectué numerosas pruebas en las que rectifiqué diversos errores en la interfaz de usuario, a su vez realicé mejoras de usabilidad simplificando los controles y modernizando la estética.

Por último, redacté las secciones y capítulos correspondientes a la parte que desarrollé del proyecto, generé la estética del documento y colaboré en los apartados de introducción y conclusiones.

9.3 Víctor Díaz Yagüe

Al comenzar el proyecto la parte que teníamos más clara era la necesidad de conocer la posición del dispositivo móvil en los pasillos del Museo García Santesmases, por lo que mi primera tarea fue investigar con la colaboración de Roberto Moreno este apartado. En dicha investigación encontramos unos proyectos realizados por alumnos de la Facultad de Informática, referenciados en el punto 3.4 de este documento.

La funcionalidad de posicionamiento de los trabajos mencionados era sencilla pero eficiente. Con un algoritmo de *K-Closest Neighbors* y el mapeo de los puntos Wifi de la facultad era suficiente para realizar una triangulación, que usaba las intensidades de los puntos de acceso Wifi.

El algoritmo estaba implementado para *Android*, pero al realizar algunas pruebas utilizando Unity como plataforma surgieron dificultades. Para solventarlas desarrollé un *plugin Android* con el que no obtuve tampoco resultados satisfactorios. Para encontrar una solución, se plantearon otros métodos como la utilización de *Beacons* pero fueron descartados porque producían el mismo problema que las soluciones anteriores.

En este punto vimos que no se podía realizar un posicionamiento *indoor* sin la utilización de *WiFi* y *Bluetooth*, por tanto, se necesitaba ubicar al usuario de otra forma. Analizando el museo y aprovechando el desarrollo del reconocimiento de imágenes que estaba generando mi compañero Roberto Moreno con la ayuda de Vuforia, decidimos posicionar al usuario con ayuda de los elementos del museo.

La funcionalidad de la aplicación sufrió diversas modificaciones a lo largo de las distintas reuniones y trabajos de análisis con el fin de mejorar la experiencia ofertada.

Ayudándome de la estructura de datos que contenía las posiciones de los objetos en base a la organización realizada, implementé un algoritmo que establecía la dirección en la que se encontraba un objeto en relación a otro. Para calcular la distancia entre ambos objetos generé un grafo en el que los vértices eran las secciones en las que estaba dividido el museo y las aristas las distancias entre ellas. Utilizando el algoritmo de *Dijkstra* calculé la distancia hasta el objeto final.

Realicé las pruebas necesarias en el museo, determinando con los resultados obtenidos que las distancias y direcciones eran correctos.

En cuanto a la memoria del proyecto, me he encargado de redactar las partes técnicas que desarrollé y a completar junto con mis compañeros el resto del documento.

Manual de instalación

En este manual se explicará la instalación en un dispositivo *Android* del software adjunto a la memoria que se ha presentado.

Cliente

El primer paso es tener disponible un dispositivo móvil con el sistema operativo Android, el software es compatible a partir de la versión 4.2 en adelante.

El segundo paso es obtener la *APK* adjunta a la memoria y transferirla del ordenador al dispositivo móvil mediante un cable *USB*.



Transferencia de datos

Una vez transferida la *APK* al dispositivo móvil se empleará un programa de gestión de archivos, por ejemplo, ES File Explorer, para buscar la *APK* insertada, seleccionarla y proceder a su instalación.

Servidor

En este apartado van a enumerar unos sencillos pasos para realizar la instalación de MongoDB en un servidor Windows.

Primero se debe descargar MongoDB para la versión del sistema operativo Windows que se necesite. Una vez descargado se procederá a su instalación, para ello se descomprime archivo descargado y posteriormente se extrae el contenido del directorio *bin* en una ruta seleccionada.

No es necesario ningún otro paso para su instalación, si se desea iniciar con los valores por defecto no es necesario configurar nada más.

Si se quiere arrancar un proceso MongoDB, bastará con abrir el CMD e ir a la ruta donde se ha descomprimido el archivo, después se debe ejecutar el siguiente comando:

```
mongod
```

Es posible que el firewall pregunte si se quiere dar permisos a la aplicación, es recomendable otorgar acceso desde el puerto por defecto, 27017.

Para comprobar que todo funciona correctamente, se tendrá que iniciar otra consola, CMD, y ejecutar el siguiente comando en la ruta donde está instalado MongoDB, esto devolverá la versión instalada.

```
./mongo  
MongoDB Shell version: 3.4.7 connecting to: test >
```

Para crear un servicio de Windows se realizarán los siguientes pasos:

Primero habrá que crear un directorio para configurar la base de datos, si fuera necesario.

```
..\mongodb> mkdir databases\db1\cfg
```





A continuación, se creará un archivo llamado **mongod.cfg** dentro del directorio anteriormente creado, en el que se insertaran las siguientes líneas:

```
logport=27666  
dbpath=..\mongodb\databases\db1
```

dbpath es la ruta donde se guardarán los archivos de la base de datos.

```
..\mongodb> ./mongod --config "..\mongodb\databases\bin\cfg\mongod.cfg" -install
```

Este comando se deberá ejecutar en un cmd como administrador, y una vez ejecutado sin errores, se tendrá un servicio llamado MongoDB que permitirá su configuración para ejecutarse de forma automática o manual.

	Módulos de creación de claves ...	El servicio IK...	Iniciado	Automático	Sistema local
	Mongo DB	Mongo DB S...		Automático	Sistema local
	Motor de filtrado de base	El Motor de ...	Iniciado	Automático	Servicio local
	Net Logon	Mantiene u...		Manual	Sistema local

Servicios Windows

Referencias

- [1] Wikipedia, enciclopedia libre. Realidad aumentada.
<https://es.wikipedia.org/w/index.php?title=Realidad_aumentada&oldid=101643132> (Fecha de consulta 2017, 3 de junio)
- [2] KIPPER, G. y RAMPOLLA, J. (2012), Augmented Reality: An Emerging Technologies Guide to AR, 1ª ed, Elsevier. ISBN: 978-1-59749-733-6
- [3] Pokemon Go. <<http://appitventures.com/successful-augmented-reality-games-makes-work/>> (Fecha de consulta 2017, 12 de junio)
- [4] Pokemon Go. <<https://www.digitaltrends.com/mobile/best-augmented-reality-apps/>> (Fecha de consulta 2017, 12 de junio)
- [5] Pokemon Go.
<<https://www.forbes.com/sites/insertcoin/2017/04/05/believe-it-or-not-pokemon-go-has-65-million-monthly-active-players/#1fe26cf2121d>> (Fecha de consulta 2017, 12 de junio)
- [6] Pokemon Go. <<https://venturebeat.com/2016/10/20/pokemon-go-is-the-fastest-mobile-game-to-hit-600-million-in-revenues/>> (Fecha de consulta 2017, 12 de junio)
- [7] Jinsha Site Museum. <<http://www.jinshasitemuseum.com/en/enabout.html>> (Fecha de consulta 2017, 14 de mayo)
- [8] Jinsha Site Museum.
<https://news.cgtn.com/news/3d4d544e786b7a4d/share_p.html> (Fecha de consulta 2017, 14 de mayo)
- [9] Jinsha Site Museum. <<https://www.pressreader.com/thailand/bangkok-post/20170203/282059096731020>> (Fecha de consulta 2017, 18 de mayo)
- [10] Kennedy Space Center.
<<http://media.kennedyspacecenter.com/kennedy/news/experience-kennedy-space-center-visitor-complex-like-never-before-with-new-advancements-ksc-smartguide-and-space-visor.htm>> (Fecha de consulta 2017, 18 de mayo)
- [11] Smithsonian Institution. <<http://naturalhistory.si.edu/exhibits/bone-hall/>> (Fecha de consulta 2017, 19 de mayo)

- [12] Smithsonian Institution. <<https://www.si.edu/exhibitions/walk-among-dinosaurs-augmented-reality-experience-5509>> (Fecha de consulta 2017, 19 de mayo)
- [13] National Museum of Singapore. <<https://www.teamlab.art/e/nms/>> (Fecha de consulta 2017, 19 de mayo)
- [14] National Museum of Singapore.
<<http://www.straitstimes.com/lifestyle/arts/national-museum-of-singapore-uses-augmented-reality-to-tell-buildings-history>> (Fecha de consulta 2017, 19 de mayo)
- [15] Caro Martínez, Marta y Hernando Hernández, David (2015), Realidad aumentada para el Museo de América. [Trabajo fin de Grado] <<http://eprints.ucm.es/32915/1/Realidad%20aumentada%20para%20el%20Museo%20de%20Am%C3%A9rica.pdf>>
- [16] Museo Thyssen-Bornemisza.
<http://www2.museothyssen.org/thyssen/app_crononautas> (Fecha de consulta 2017, 23 de mayo)
- [17] Comparativa de plataformas de realidad aumentada.
<<http://socialcompare.com/en/comparison/augmented-reality-sdks>> (Fecha de consulta 2017, 14 de agosto)
- [18] Artoolkit. <<https://artoolkit.org/>> (Fecha de consulta 2017, 20 de junio)
- [19] Artoolkit. <<https://en.wikipedia.org/wiki/ARToolKit>> (Fecha de consulta 2017, 20 de junio)
- [20] NyArtoolkit. <http://nyatla.jp/nyartoolkit/wp/?page_id=198> (Fecha de consulta 2017, 20 de junio)
- [21] ATOMIC. <https://en.wikipedia.org/wiki/ATOMIC_Authoring_Tool> (Fecha de consulta 2017, 20 de junio)
- [22] Augmented. <<http://augmentedpixels.com/>> (Fecha de consulta 2017, 22 de junio)
- [23] Augmented. <<https://www.crunchbase.com/organization/augmented-pixels-co#/entity>> (Fecha de consulta 2017, 22 de junio)
- [24] EasyAR. <<https://www.easyar.com/>> (Fecha de consulta 2017, 29 de junio)

- [25] Kudan. <<https://www.kudan.eu/>> (Fecha de consulta 2017, 29 de junio)
- [26] Kudan. <<https://www.crunchbase.com/organization/kudan#/entity>> (Fecha de consulta 2017, 29 de junio)
- [27] Wikitude. <<https://www.wikitude.com/>> (Fecha de consulta 2017, 1 de julio)
- [28] Wikitude. <<https://en.wikipedia.org/wiki/Wikitude>> (Fecha de consulta 2017, 1 de julio)
- [29] Vuforia. <<https://www.vuforia.com/>> (Fecha de consulta 2017, 7 de agosto)
- [30] Vuforia. <https://en.wikipedia.org/wiki/Vuforia_Augmented_Reality_SDK> (Fecha de consulta 2017, 7 de agosto)
- [31] Álvarez Sánchez, Christian y Arboleda Arturo, Kevin Darío (2016), Generación de historias centradas en el usuario con plataformas móviles [Trabajo fin de Grado] <<http://eprints.ucm.es/38676/1/ParableMemoria.pdf>>
- [32] López Mañas, Enrique y Moreno Nacarino, y Plá Herrero, Javier (2010), Proyecto Avanti: sistema de asistencia a la evacuación de incendios [Trabajo de curso] <<http://eprints.ucm.es/11048/1/MemoriaAVANTI.pdf>>
- [33] Unity. <<https://unity3d.com/es/>> (Fecha de consulta 2017, 7 de agosto)
- [34] Unity. <[https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine))> (Fecha de consulta 2017, 9 de agosto)
- [35] Unity & Vuforia. <<https://www.sitepoint.com/how-to-build-an-ar-android-app-with-vuforia-and-unity/>> (Fecha de consulta 2017, 9 de agosto)
- [36] Radkowski, Rafael y Oliver, James H. (2013), Natural Feature Tracking Augmented Reality for On-Site Assembly Assistance Systems [Iowa State University]<http://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=1134&context=me_pubs>
- [37] MongoDB. <<https://www.mongodb.com/es>> (Fecha de consulta 2017, 7 de abril)